MOOC课程配套教材

数字电子技术基础

Fundamentals of Digital Electronics

◎陈新龙　主　编
何　伟　副主编

清華大學出版社

数字电子技术基础

陈新龙　主　编
何　伟　副主编

清华大学出版社
北　京

内容简介

本书以主编已出版的4套电工电子技术国家级规划教材为基础，从黑白帽子逻辑问题引出逻辑运算基础知识；从分立元器件、TTL集成逻辑门、CMOS集成逻辑门、硬件描述语言4个方面介绍了常用逻辑运算的电路实现；讲解了组合逻辑电路、时序逻辑电路的分析设计方法，典型电路构成特点及集成芯片的逻辑功能与应用；阐述了脉冲电路及存储器、A/D转换器、D/A转换器等大规模集成电路的特点；最后介绍了利用Verilog HDL描述数字逻辑电路的方法。

本书为“卡通说解数字电子技术”MOOC配套教材，各知识点均配有讲解视频，可微信扫码书中二维码在线观看视频。“卡通说解数字电子技术”MOOC基于虚拟教室以一张PPT阐述清楚一个知识点的思路建设了113个教学视频，利用图片映射题、基于图形的文字填空题等形式建设在线习题近300道。在线习题支持智能辅导、自动批阅，有利于更牢固地掌握各知识点，非常适合以翻转课堂教学形式开展教学活动，可作为电气类、电子信息类等大类专业的“数字电子技术”“数字电路”及类似课程的教材。

图书在版编目(CIP)数据

数字电子技术基础/陈新龙主编，何伟副主编. —北京：清华大学出版社，2018
ISBN 978-7-302-50627-0

Ⅰ. ①数…　Ⅱ. ①陈…　②何…　Ⅲ. ①数字电路—电子技术　Ⅳ. ①TN79

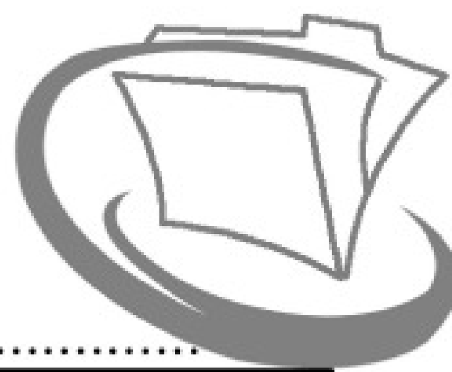
中国版本图书馆CIP数据核字(2018)第154753号

责任编辑：文　怡
封面设计：台禹微
责任校对：李建庄
责任印制：刘海龙

出版发行：清华大学出版社
　网　　址：http://www.tup.com.cn，http://www.wqbook.com
　地　　址：北京清华大学学研大厦A座　　**邮　　编**：100084
　社 总 机：010-62770175　　**邮　　购**：010-62786544
　投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
　质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
　课件下载：http://www.tup.com.cn，010-62795954
印 装 者：三河市少明印务有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：23.75　　**字　　数**：575千字
版　　次：2018年9月第1版　　**印　　次**：2018年9月第1次印刷
定　　价：59.00元

产品编号：074817-01

前言



FOREWORD



片花

电的发现是人类社会最伟大的发现之一。电子的流动是一种能量的流动，在带给人们光明与动力的同时推动了一个时代的进步。半导体器件的出现赋予电子的流动以新的内涵。半导体器件的应用使这种能量的流动成为一种信号的传递，一种超强功能的集成信息的传输。集成电路的问世引起了电子技术领域一场新的革命，超大规模集成电路的深入应用推动着一个新时代的来临。在这个时代里，数字电子技术无处不在，因此，各大高校电气类、电子信息类各专业均开设了“数字电子技术”“数字电路”或类似课程。

必须指出，“数字电子技术”是一门探讨数字逻辑、研究电子器件及其应用的课程，其理论性、专业性、应用性均较强。数字电子技术领域新型电子器件不断涌现，基于新型电子器件的设计方法不断推陈出新。如何在规定的学时数内使学生掌握逻辑运算的基础知识，理解典型数字逻辑电路器件逻辑功能及其应用方法，跟随新型电子器件的应用模式，成为教学实施的难点。

本书从黑白帽子逻辑问题引出逻辑运算基础知识；从分立元器件、TTL 集成逻辑门、CMOS 集成逻辑门、硬件描述语言 4 个方面介绍了常用逻辑运算的电路实现；从逻辑模型、Quartus Ⅱ仿真两个方面介绍了组合逻辑电路、时序逻辑电路的分析方法；讲解了组合逻辑电路、时序逻辑电路的 SSI 设计方法，重点介绍了典型电路的构成特点及集成电子器件的逻辑功能与应用；阐述了脉冲电路及存储器、A/D 转换器、D/A 转换器等大规模集成电路的特点；最后介绍了利用 Verilog HDL 描述数字逻辑电路的方法。

本书继承了主编已出版的 4 套电工电子技术国家级规划教材的建设成果，力图通俗易懂。本书编写时相对压缩了数字电子技术各基础理论，淡化了电子器件内部电路的分析，强调了电子器件的逻辑特点及其应用方法，在强调掌握数字逻辑电路基本分析方法的前提

下，突出了利用EDA软件仿真分析数字逻辑电路，嵌入了Quartus Ⅱ计算机仿真结果，以使读者快速理解重点研究的电子器件的逻辑特点。

本书为在线教学版教材，建设有配套的在线课程，课程名为“卡通说解数字电子技术”，各知识点均配有卡通形式的讲解视频，可微信扫描书中二维码在线观看视频。

在线课程“卡通说解数字电子技术”基于虚拟教室，以一张PPT阐述清楚一个知识点的思路建设了113个教学视频，利用图片映射题、基于图形的文字填空题等习题形式建设在线习题近300道。在线习题支持智能辅导、自动批阅，有利于更牢固地掌握各知识点，非常适合于以翻转课堂教学形式开展教学活动。

中国大学MOOC平台“卡通说解数字电子技术”网址：

https://www.icourse163.org/course/CQU-1002532002

学堂在线MOOC平台“卡通说解数字电子技术”网址：

http://www.xuetangx.com/courses/course-v1:CQU+ICE20012+sp/about

本书编写时参照了教育部高等学校电子信息科学与电气信息类基础课程教学指导分委员会于2004年12月制定的《数字电子技术基础教学基本要求》，可作为电气类、电子信息类等大类专业的“数字电子技术”“数字电路”及类似课程的教材。

本书中元器件符号尽量与仿真软件保持一致，便于读者应用。

本书第2～4章由何伟主持编写，第10章及附录由胡国庆整理编写，其余各章由陈新龙整理编写。全书由陈新龙担任主编，何伟担任副主编。在本教材的建设过程中，得到了重庆大学第2批在线课程建设项目的支持。此外，重庆大学国家级教学名师曾孝平教授、重庆大学通信工程学院黄智勇副教授以及许多老师和同学均对本书提出了宝贵的、建设性的意见与建议，在此表示感谢。

由于作者水平有限，疏漏和不妥之处在所难免，敬请读者批评指正。主编邮箱为clxtx@cqu.edu.cn，副主编邮箱为1704000@qq.com。

作　者

2018年6月

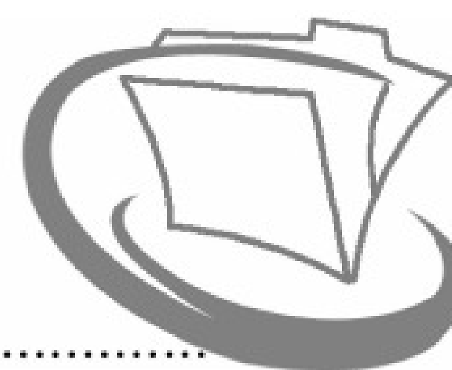


本书课件下载

常 用 符 号

符 号	描 述	符 号	描 述
A、B、C、D	逻辑输入量	E	电动势
F、Y	逻辑输出量	f	频率
G	电导	R、C	电阻、电容
R_C	三极管集电极电阻	R_B、R_E	三极管基极、发射极电阻
R_G	场效应管栅极电阻	R_D	场效应管漏极电阻
R_S	场效应管源极电阻、信号源内阻	R_f	反馈电阻
I	直流电流、正弦电流有效值	i	交流电流
I_{IH}	输入为高电平时的输入电流	I_{IL}	输入为低电平时的输入电流
I_{OH}	输出为高电平时的输出电流	I_{OL}	输出为低电平时的输出电流
I_F	二极管最大整流电流、反馈电流信号	I_R	二极管反向电流
I_B、I_E	基极、发射极电流	I_{BQ}、I_{EQ}	基极、发射极静态电流
I_{BS}、I_{ES}	基极、发射极临界饱和电流	I_{CQ}	集电极静态电流
I_{CEO}	集电极与发射极之间的反向截止电流（穿透电流）	I_C、I_{CS}	集电极电流、三极管集电极临界饱和电流
U	直流电压、正弦电压有效值	u	交流电压
U_I	输入电压	U_O	输出电压
U_{CC}	三极管工作电源	U_{DD}	MOS管工作电源
U_{OH}	输出高电平	U_{OL}	输出低电平
U_{ON}	开门电平	U_{OFF}	关门电平
U_{OHMIN}	输出高电平的最小值	U_{IHMIN}	输入高电平的最小值
U_{OLMAX}	输出低电平的最大值	U_{ILMAX}	输入低电平的最大值
U_{NH}	高电平噪声容限	U_{NL}	低电平噪声容限
U_{TH}	门槛电平	U_m	脉冲幅度
U_{T+}	接通电位	U_{T-}	断开电位
U_D	二极管压降	$U_{(BR)}$	PN结反向击穿电压
U_R	二极管最大反向工作电压	u_T	温度电压当量
U_{CES}	三极管集电极、发射极间临界饱和电压	U_{CE}	三极管集电极、发射极间电压
U_{BE}	三极管发射极电压	U_{CB}	三极管集电极电压
U_{BB}	三极管基极直流电压源电压	U_{CC}	三极管集电极直流电压源电压
$U_{BE(ON)}$	三极管发射结导通压降	U_Z	稳压管稳定电压
U_{GS}	场效应管栅、源极间电压	U_{DS}	场效应管漏、源极间电压
$U_{GS(th)}$	开启电压	$U_{GS(off)}$	夹断电压
U_{GD}	场效应管栅、漏极间电压	U_{ON}	二极管正向导通压降
N_O	扇出系数	ΔU_T	回差电压
TG	CMOS传输门	β、h_{fe}	三极管电流放大系数
t_{pd1}、t_r	上升延迟时间	t_{pd2}、t_f	下降延迟时间
t_{pd}、t_r	平均传输延迟时间	t_w	脉冲宽度
D、D_Z	二极管、稳压二极管	T	三极管

目 录

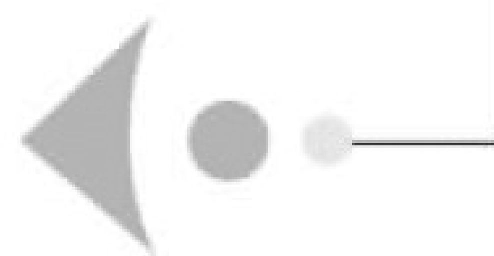


CONTENTS

标有星号(*)的小节属于非必修内容,可以选讲或选修。



第1章



绪论

本章要点：

本章为本书基础章，从数字量与逻辑量的概念出发，介绍数字信号、数字电路、数字电子技术等基础概念；介绍数制、码制的基础知识；从方便电路实现角度，介绍数字系统中固定位数二进制数的表示方法；基于固定位数二进制数的表示方法，介绍数字系统中通过补码用加法实现减法的方法。

1.1 概述



自然界的数据有模拟、数字 2 种类型，理解数字量是学习本书的基础。

1.1.1 数字量与逻辑量

自然界的绝大多数物理量为时间上和数量上均连续变化的物理量，称为模拟量，如声音、压力等。表征这些模拟量的信号称为模拟信号，相应的数据称为模拟数据。

另外，自然界中还有一些数据是不连续变化的，如人的个数，这些数据量称为数字量。在这些数字量中，还有一些特殊的量只有两种值，如河南安阳发现的曹操墓是否为真的曹操墓，其值只有"是""不是"两种("不知道"是指不知道其值是"是"还是"不是"，不是一种值)，把这种只有两种值的量称为逻辑量。理解逻辑量是掌握逻辑运算的基础。

基于逻辑量，人们构建了很多经典问题，用于锻炼、测试逻辑推理能力。例如：

有 3 顶黑帽子和 2 顶白帽子，让 3 个人从前到后站成一排，每个人都看不见自己戴的帽子的颜色，只能看见站在前面那些人的帽子颜色。给每个测试者头上戴一顶黑帽子，然后从后到前逐个问测试者所戴帽子的颜色，最后者和中间者均回答说"不知道"，请问最前面的人是如何回答的？

答案如下：

最前面的人很自信地回答："黑色！"

推理过程如下：

最后面的人知道自己帽子颜色的前提是前面 2 个人戴的都是白帽子，基于只有 2 顶白帽子的已知条件，可推出自己戴的是黑帽子。本例中，最后面的人回答"不知道"，说明最前面 2 个人戴的不全是白帽子。

中间的人不知道自己戴的帽子颜色，说明最前面的人戴的不是白帽子。

基于上面的推理，最前面的人可推出自己戴的是黑帽子。

1.1.2 数字信号与数字电子技术

1849 年，英国数学家乔治·布尔(George Boole)总结提出逻辑代数，也称为布尔代数。逻辑代数中，变量及函数的取值只能取逻辑 0 和逻辑 1(yāo)两种不同的逻辑状态，便于电路实现。例如，用高电平(有电流、开关闭合)表示 1，低电平(无电流、开关断开)表示 0，这种表示方式称为正逻辑；反之，称为负逻辑。在本书中，不加说明均指正逻辑。

利用逻辑代数，可解决现实世界中的信号处理及其运算问题，前提条件是这些信号是用 0 和 1 表示的序列，是时间上和数值上都不连续变化的物理量，是数字信号。

数字信号只有 0 和 1 两个符号，数值上的不连续显而易见。另外，数字信号在时间上也是不连续的，在固定的时间间隔内，数字信号只能有 1 个值，一般用同步时间脉冲来体现固定的时间间隔。例如，主频为 800MHz 的智能手机运算极限是每秒处理 800M 个 0 或 1 的符号。

工作在数字信号下的电路称为数字电路，利用数字电路解决现实世界中问题的技术统称为数字电子技术。

当然，现实世界中的常见信号均为模拟信号，不能直接送入数字系统进行处理，需要将模拟信号离散并数字化为数字信号。模拟信号的离散化过程如图 1.1.1 所示。

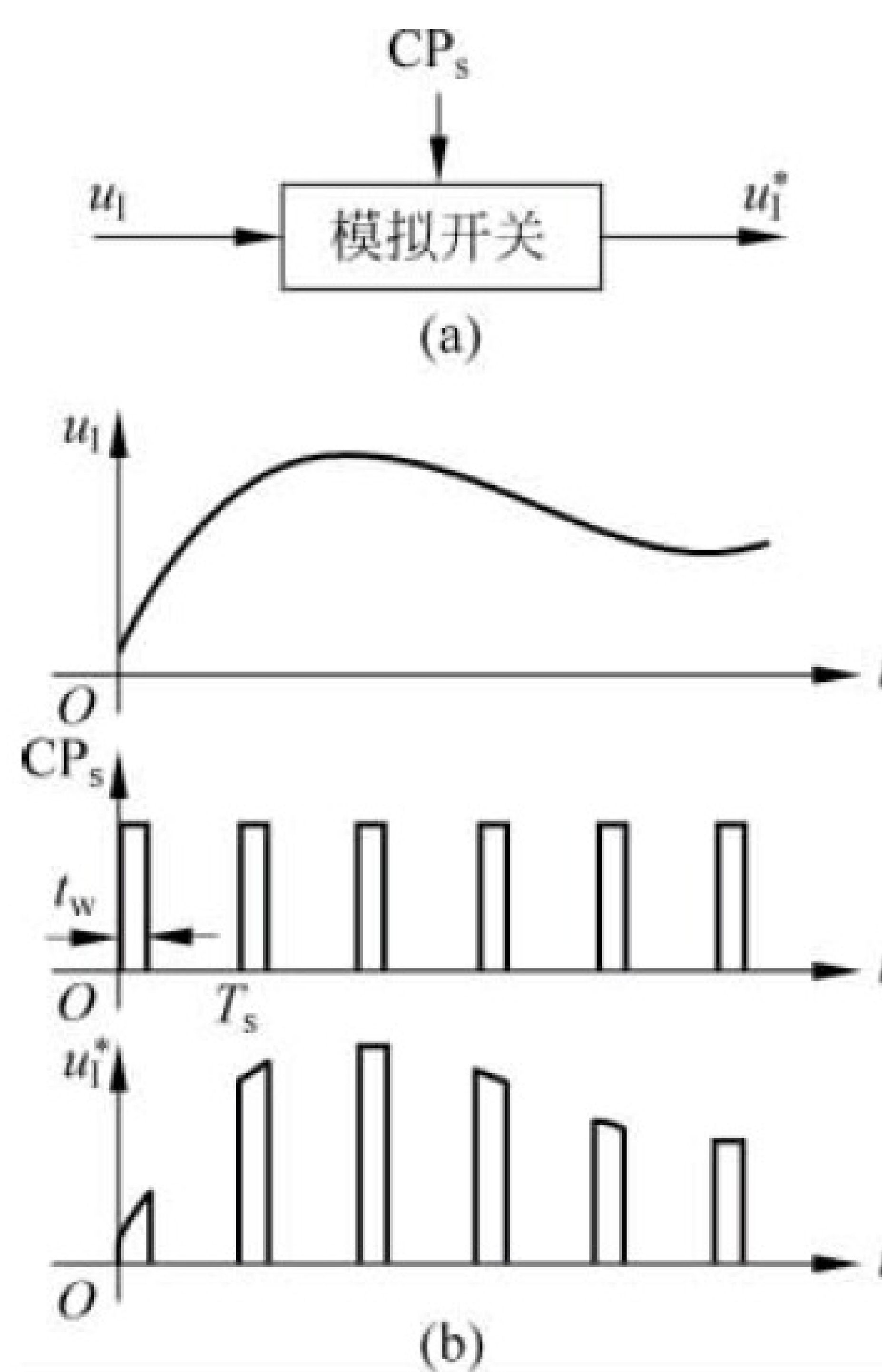


图 1.1.1 模拟信号的离散化

1.1.3 学习"数字电子技术"的方法

"数字电子技术"是一门大类基础课，要学习该课程首先必须理解该课程的基础知识。

众所周知，加、减、乘、除是初等数学的基本运算，其他运算均由这 4 种基本运算组合而成。逻辑代数是分析与设计数字电路的数学工具，也有其基本运算。

在上面的黑白帽子逻辑问题中，最前面的人能很自信地回答"黑色"的原因主要基于以下几种逻辑联系：

(1) 帽子颜色只有黑、白两种,若不是黑帽子,则必是白帽子。

(2) 白帽子只有 2 顶,3 人中必有黑帽子。

(3) 最后者和中间者均回答"不知道"。

基于逻辑联系(1)可抽象出非逻辑运算,基于逻辑联系(2)和(3)可概括总结出与、或两种基本逻辑运算,后面将会介绍这 3 种基本逻辑运算的含义,在此仅指出,逻辑表达式由且只能由与、或、非 3 种基本逻辑运算组成。

当然,逻辑代数也能实现加、减、乘、除等数学运算问题,只是这些运算由且只能由与、或、非 3 种基本逻辑运算组成,也可以说,数学运算通过逻辑运算来实现。

当然,逻辑运算是通过电路来实现的,要学习本课程,必须具有一定的电路基础知识。

基于上面的分析,可总结出学习本课程的基本方法:温习以前学过的电路基础知识,扎实掌握本课程的基础知识——逻辑运算基础。

复习与思考

1.1.1 下列描述中不正确的是( )。

(1) 逻辑量只有两种值

(2) 用逻辑 0 表示低电平,逻辑 1 表示高电平,这种逻辑表示方法称为负逻辑

(3) 对于 1 和 0 可用电位的高和低或用脉冲信号的有和无来表示

(4) 逻辑运算由与、或、非 3 种基本运算及其组合构成

1.1.2 如图 1.1.2 所示,描述更贴切的是( )。



图 1.1.2 复习与思考 1.1.2 的图

1.2 数制与码制

理解逻辑运算首先必须理解数学上数的表示方法,掌握数制是学习本课程的基础。

1.2.1 常见数制



表示一个数习惯上采用位置记数法,包括数码、基数、位权 3 个要素,具体如图 1.2.1 所示。

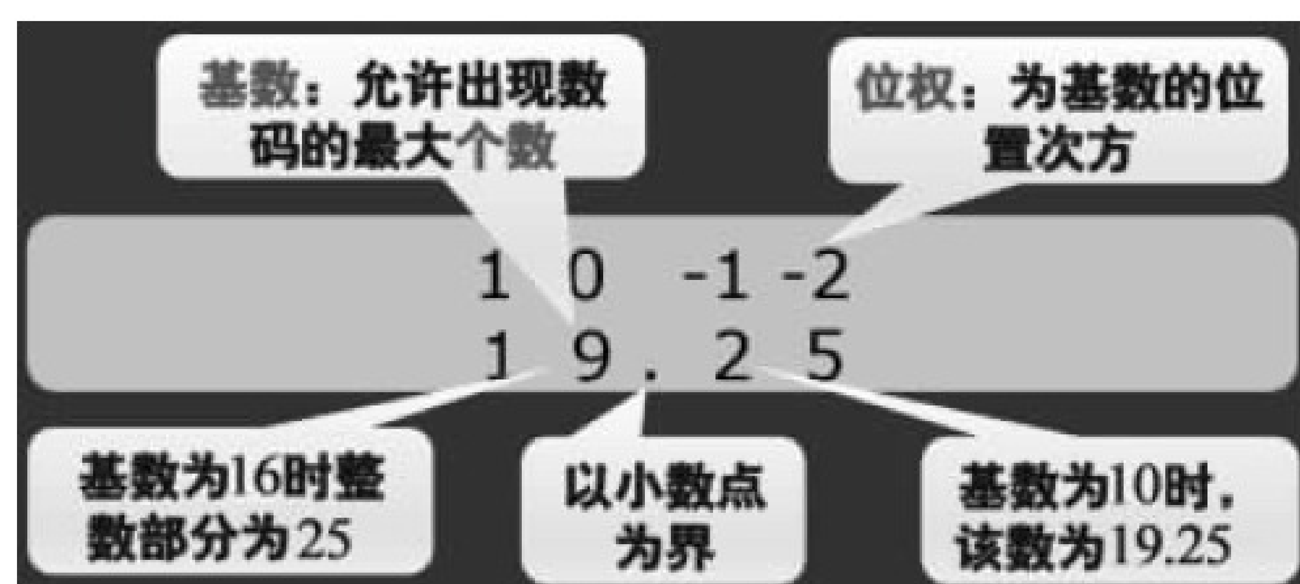


图 1.2.1　位置记数法

数码为构成该计数进制的具体数码，基数为允许出现数码的最大个数，位权为基数的位置次方。基于这 3 个要素，可总结几种常见进制如下。

1. 十进制

十进制包括 0、1、2、3、4、5、6、7、8、9 十个数码，基数为 10。这种按照"逢十进一"进行计数的计数方法称为十进制。

在十进制数中，它的计数规律是"逢十进一"，即 $9+1=10$。在"10"中，右边的"0"为个位数，左边的"1"为十位数，也就是 $10=1\times10^1+0\times10^0$。这样一来，每一数码处于不同的位置时（数位），它代表的数值是不同的。例如，数 224.36 可以写成

$$224.36=2\times10^2+2\times10^1+4\times10^0+3\times10^{-1}+6\times10^{-2}$$

从电路的角度来看，采用十进制是不方便的。因为构成电路的基本想法是把电路的状态和数码对应起来，而十进制的 10 个数码，必须由 10 个不同的而且能严格区分的电路状态与之对应，这将在技术上带来许多困难，而且很不经济。所以，在数字电路中不直接采用十进制。

2. 二进制

前面提到，数学运算通过逻辑运算来实现，要用逻辑运算来实现数学运算首先应将数学运算表示成 0 和 1 的序列。利用 0、1 两个数码构成数据，按照"逢二进一"进行计数的计数方法称为二进制。

在二进制数中，有 0、1 两个数码，基数为 2，计数规律是"逢二进一"，即 $1+1=10$。在"10"中，右边的"0"为 2^0 位数，左边的"1"为 2^1 位数，也就是 $10=1\times2^1+0\times2^0$。因此，每一数码处于不同的位置时（数位），它代表的数值是不同的。例如，数 11010.01 可以写成

$$11010.01=1\times2^4+1\times2^3+0\times2^2+1\times2^1+0\times2^0+0\times2^{-1}+1\times2^{-2}$$

从电路实现的角度，二进制具有许多优点，因此在数字电子技术中广泛采用二进制。二进制只有 0、1 两个数码，它的每一位数都可以用任何具有两个不同稳定状态的元件来表示，所以电路简单、可靠，所用元件少。二进制的基本运算规则简单，操作简便，便于电路实现。

3. 十六进制和八进制

十六进制包括 0、1、2、3、4、5、6、7、8、9、A(对应十进制数中的 10)、B(11)、C(12)、D(13)、E(14)、F(15)十六个数码,基数为 16。这种按照"逢十六进一"进行计数的计数方法称为十六进制。

按照"逢十六进一"进行计数的数称为十六进制数,是以 16 为基数的计数体制。例如,数$(63.A)_{16}$可以写成

$$(63.A)_{16}=6\times16^1+3\times16^0+A\times16^{-1}$$

八进制包括 0、1、2、3、4、5、6、7 八个数码,基数为 8。这种按照"逢八进一"进行计数的计数方法称为八进制。

按照"逢八进一"进行计数的数称为八进制数,是以 8 为基数的计数体制。例如,数$(37.5)_8$可以写成

$$(37.5)_8=3\times8^1+7\times8^0+5\times9^{-1}$$

1.2.2 数制间的转换

尽管二进制数用电路实现简单,但人们只习惯十进制数。下面介绍各种数制间的转换。

1. "二—十"转换

将二进制数转换为等值的十进制数称为"二—十"转换。可按二次幂相加法进行转换。例如

$$(1011.01)_2=1\times2^3+0\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}=(11.25)_{10}$$

2. "十—二"转换

将十进制数转换为等值的二进制数称为"十—二"转换,可通过一个例题来理解。

【例 1.2.1】 将十进制数 25.375 转换为二进制数。

解 将带小数的非二进制数转换为二进制数应将整数、小数部分单独转换。

(1) 对整数部分用辗转除 2 取余法。

$$0\xleftarrow{\div2}1\xleftarrow{\div2}3\xleftarrow{\div2}6\xleftarrow{\div2}12\xleftarrow{\div2}25$$

高位	1	1	0	0	1	低位

所以

$$(25)_{10}=(11001)_2$$

(2) 对小数部分用辗转乘2取整法。

$$\begin{array}{ccccccc} 0.375 & \xrightarrow{\times 2} & 0.75 & \xrightarrow{\times 2} & 0.5 & \xrightarrow{\times 2} & 0 \\ \downarrow & & \downarrow & & \downarrow & & \\ \text{负的低位}\ 0 & & 1 & & 1\ \text{负的高位} & & \end{array}$$

所以

$$(0.375)_{10}=(0.011)_2$$

$$(25.375)_{10}=(11001.011)_2$$

3. "二—八(或十六)"转换

将二进制数转换为八进制数(或十六进制数)称为"二—八(或十六)"转换,可通过一个例题来理解。

【例 1.2.2】 将$(10011100101101001000.1001)_2$转换为八进制数和十六进制数。

解 (1) 转换为八进制数。将二进制数转换为八进制数的方法如下:从小数点开始,整数部分向左、小数部分向右每3位二进制数分为一组,对应1位八进制数。即

$$\begin{array}{cccccccccc} \underline{010} & \underline{011} & \underline{100} & \underline{101} & \underline{101} & \underline{001} & \underline{000} & . & \underline{100} & \underline{100} \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & & \downarrow & \downarrow \\ 2 & 3 & 4 & 5 & 5 & 1 & 0 & . & 4 & 4 \end{array}$$

所以

$$(10011100101101001000.1001)_2=(2345510.44)_8$$

(2) 转换为十六进制数。将二进制数转换为十六进制数的方法如下:从小数点开始,整数部分向左、小数部分向右每4位二进制数分为一组,对应于1位十六进制数。即

$$\begin{array}{ccccccc} \underline{1001} & \underline{1100} & \underline{1011} & \underline{0100} & \underline{1000} & . & \underline{1001} \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & & \downarrow \\ 9 & C & B & 4 & 8 & . & 9 \end{array}$$

所以

$$(10011100101101001000.1001)_2=(9CB48.9)_{16}$$

4. "八—十六"转换

【例 1.2.3】 将$(753.234)_8$转换为十六进制。

解 (1) 先将八进制转换为二进制数。将八进制数转换为二进制数的方法如下:从小数点开始,整数部分向左、小数部分向右每1位八进制数对应3位二进制数。即

$$\begin{array}{ccccccc} 7 & 5 & 3 & . & 2 & 3 & 4 \\ 111 & 101 & 011 & . & 010 & 011 & 100 \end{array}$$

所以

$$(753.234)_8=(111101011.0100111)_2$$

(2) 将对应的二进制数转换为十六进制数。

$$\begin{matrix} 0001 & 1110 & 1011. & 0100 & 1110 \\ 1 & E & B. & 4 & E \end{matrix}$$

所以

$$(753.234)_8=(1EB.4E)_{16}$$

根据上面的方法,可类似总结十六进制数到八进制数的转换方法:先将十六进制数转换为二进制数,转换时从小数点开始,整数部分向左、小数部分向右每1位十六进制数对应4位二进制数。然后,将对应的二进制数转换为八进制数即可。

1.2.3 码制



在数字电子技术中,用于数值计算的数据采用二进制表示。在许多场合,除了数值计算以外,常常需要进行文字、符号等信息的处理。用二进制序列表示文字、符号等信息的编码方法称为码制。

显然,码制与数制是两个不同的概念,同样的二进制序列,如果是码制,则是一些约定的符号,如果是用于计算的数据,则可用具体的十进制数表示。

使用二进制数码进行编码的方式很多,在本书中仅介绍BCD码、格雷码、ASCII码。

1. 二—十进制代码(BCD码)

用二进制数码表示1位十进制数的0～9的10个状态的编码称为BCD(Binary-Coded Decimal)码。显然,0～9的10个十进制数码需要用4位二进制码才能表示。其编码方法很多,最常见的BCD码如表1.2.1所示。

表1.2.1 几种常见的BCD码

数 码	编 码		
	8421码	2421码	余3码
0	0 0 0 0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 0 1 0	0 1 0 1
3	0 0 1 1	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 1 1	1 0 0 0
6	0 1 1 0	1 1 0 0	1 0 0 1
7	0 1 1 1	1 1 0 1	1 0 1 0
8	1 0 0 0	1 1 1 0	1 0 1 1
9	1 0 0 1	1 1 1 1	1 1 0 0
权	8 4 2 1	2 4 2 1	

在上述编码中,最常用的编码为8421码。8421码属于有权码,即每一位都有固定的权值。从左到右,每一位的1分别表示8、4、2、1,将每一位的1代表的十进制数加起来,得到的结果就是它所代表的十进制数码。所以这种代码称为8421码。类似地,2421码每一位的权值从左到右分别为2、4、2、1,将每一位的1代表的十进制数加起来,得到的结果就是它所代表的十进制数码。

余3码的编码规则与8421码不同,如果把每一个余3码看作4位二进制数,则它的数值要比它所表示的十进制数码多3,故将这种代码称为余3码。余3码属于无权码,即每一位无固定权值。

2. 格雷码

在一组数的编码中,若任意两个相邻的代码只有1位二进制数不同,则称这种编码为格雷码(Gray Code)。格雷码又称循环码或反射码,其最大数与最小数之间仅1位数不同,即“首尾相连”。

4位自然二进制码与4位典型格雷码的对照如表1.2.2所示。

表1.2.2 二进制码与格雷码

编码顺序	二进制码	格雷码
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 1
3	0 0 1 1	0 0 1 0
4	0 1 0 0	0 1 1 0
5	0 1 0 1	0 1 1 1
6	0 1 1 0	0 1 0 1
7	0 1 1 1	0 1 0 0
8	1 0 0 0	1 1 0 0
9	1 0 0 1	1 1 0 1
10	1 0 1 0	1 1 1 1
11	1 0 1 1	1 1 1 0
12	1 1 0 0	1 0 1 0
13	1 1 0 1	1 0 1 1
14	1 1 1 0	1 0 0 1
15	1 1 1 1	1 0 0 0

3. ASCII码

ASCII(American Standard Code for Information Interchange,美国信息交换标准代码)是基于拉丁字母的一套计算机编码系统,是目前通用的单字节编码系统,也是国际上通用的字符码,主要用于显示现代英语和其他西欧语言。

ASCII码的编码如表1.2.3所示。

表 1.2.3 ASCII 码的编码

ASCII 值	控制字符	ASCII 值	控制字符	ASCII 值	控制字符	ASCII 值	控制字符
0	NUT	32	(space)	64	@	96	、
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(	72	H	104	h
9	HT	41	)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	—	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DCI	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	X	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	TB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[	123	{
28	FS	60	<	92	\	124	|
29	GS	61	=	93	]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	__	127	DEL

复习与思考

1.2.1 对于数制,下列描述中不正确的是(　　)。

(1) 表示一个数习惯上采用位置记数法,包括数码、基数、位权 3 个要素

(2) 八进制数 129 对应十进制的值为 89

(3) 当将十进制数整数部分转换为二进制时,可采用除 2 取余法

(4) 以上答案至少有 1 个不正确

1.2.2 请将十进制数 105.34 转换为二进制,保留小数点后 6 个有效位,写出转换结果,讨论转换后的数据和原始数据转换精度方面的差异。

1.3 固定位数二进制数的算术运算

数字系统中采用固定位数二进制数表示数据，因此，理解固定位数二进制数的运算特点是学习本课程的基础。

1.3.1 二进制数的算术运算

二进制数算术运算与十进制数算术运算规则相同，可通过几个例题来理解。

【例 1.3.1】 用二进制数完成十进制数 11_D+25_D的运算。

解 $11_D=1011$　　$25_D=11001$

运算过程如下：

$$\begin{array}{r} 1\,0\,1\,1 \\ +\ 1\,1\,0\,0\,1 \\ \hline 1\,0\,0\,1\,0\,0 \end{array} = 36_D$$

【例 1.3.2】 用二进制数完成十进制数 25_D-11_D的运算。

解 $25_D=11001$　　$11_D=1011$

运算过程如下：

$$\begin{array}{r} 1\,1\,0\,0\,1 \\ -\ 1\,0\,1\,1 \\ \hline 0\,1\,1\,1\,0 \end{array} = 14_D$$

有兴趣的读者可对照十进制数进一步进行二进制数的乘法和除法运算。

1.3.2 固定位数二进制数的表示方法及其运算

顾名思义，采用固定位数表示二进制数的方法为规定二进制数的具体位数，如 16 位、32 位等，超过规定位数的数据部分视为溢出，只截取规定位数的二进制数作为数据。对于固定位数的二进制数，其数值为有限大小，而数学上，数可以接近无限大小。

因为二进制数据位数固定，因此存在着最高数据位，可方便表示负数。对于有符号数，数据位不变，规定最高位为符号位。最高位为 0 表示正数，最高位为 1 表示负数，这种表示方法称为二进制数的原码表示法。当然，也可约定二进制数为无符号数，最高位纳入数据位。

下面，以 5 位固定二进制位数的有符号数为例进行说明。

【例 1.3.3】 用二进制数完成十进制数 4_D+3_D的运算。

解 两个数均为正数，固定 5 个二进制位，最高位为符号位，有

$$4_D=00100 \qquad 3_D=00011$$

运算过程如下：

$$\begin{array}{r} 00100 \\ +\ 00011 \\ \hline 00111 \end{array} = 7_D$$

【例 1.3.4】 用二进制数完成十进制数 $4_D - 3_D$的运算。

解　　　　$4_D = 00100$　　　$-3_D = 10011$

运算过程如下：

$$\begin{array}{r} 00100 \\ +10011 \\ \hline 10111 \end{array} = -7_D$$

从上面的运算结果可以看出，将一个正数和一个负数的原码直接相加，结果不正确。此外，两个同符号数相加时，它们的绝对值不可超过有效数字位所能表示的最大值，否则，也会得出错误的结果。

1.3.3　固定位数二进制数用加法实现减法的原理及方法

在计算机等数字系统中，只有加法器一个运算设备，减法通过加法实现。当然，例 1.3.2 显示，用一个正数的原码直接加上一个负数的原码，结果不正确，那么，计算机中是如何实现减法的呢？

计算机中的数据采用固定的二进制位数表示，是有限值，达到理论最大值时将产生溢出（归零），这也是 24 点和 0 点是同一个时间点的道理所在。

基于这个特点，假定计算机中最大值为 12，对于 $4+9=13$，按照最大值归零的道理，有

$$4+9=13-12=1=4-3$$

可见，当最大值为 12 时，$4-3$ 等于 $4+9$。

基于上面的分析，当数据采用固定的二进制位数表示时，一个数减去另一个数，可用这个数加上另一个数来实现。

当然，采用固定 n 位表示二进制数并不会出现最大值为 12 的情况，这个理论最大值只能是 2^n。采用上面的例子进行说明，是为了方便读者理解。

可以对照老式闹钟理解 $4-3$ 等于 $4+9$。假定当前时间为 4 点，将时钟回拨 3 个刻度（相当于 $4-3$），当前时间将设置为 1 点。将时钟往前拨 9 个刻度（相当于 $4+9$），当前时间也将设置为 1 点。可见，当最大值为 12 时，$4-3$ 等于 $4+9$。

基于上面的实例可总结，数 a 减去数 b 时，可用数 a 加上另一个数 c 来实现，c 和 b 的关系为

$$c=\text{最大值}-b$$

可见，当一个正数加上一个负数时，可用这个正数加上最大值减去负数的绝对值来实现，从而用加法实现减法。

可能有人会说,“c=最大值-b”依然是减法。没错,“最大值-b”依然是减法,但最大值是一个特殊的数,因此按照最大值归零的道理,有

$$c=最大值-b=0-b$$

【例 1.3.5】 假定固定 5 个二进制位表示数据,请不考虑符号位求 0_D-3_D的运算结果。

解 不考虑符号位,两个数用 4 个二进制位表示,有

$$0_D=0000 \qquad -3_D=0011$$

运算过程如下:

$$\begin{array}{r} 0\ 0\ 0\ 0 \\ -\ \ \underset{\cdot}{0}\ \underset{\cdot}{0}\ \underset{\cdot}{1}\ \underset{\cdot}{1} \\ \hline 1\ 1\ 0\ 1 \end{array}=13_D$$

固定 5 个二进制位表示数据,除掉符号位,最大值为 $2^4=16$,$0-3=13=16-3$,结果正确。

仔细观察 3 和 13 的二进制位可知,13 的值等于 3 的各二进制位逐位取反后加 1。

可见,尽管“最大值-b”依然是减法,但是可用 b 的各二进制位逐位取反(称为反码)加 1 来求得,而无须用减法求得。

把二进制数的这种表示方法形象地称为补码。具体而言,正数的补码等同于原码,负数的补码为符号位不变,其余各二进制位逐位取反后加 1。

固定位数二进制数用加法实现减法的方法如下:加数、被加数均用补码表示,将加数、被加数直接相加即可求得正确结果,当然,结果依然是补码。

【例 1.3.6】 假定固定 5 个二进制位表示数据,请结合补码求 4_D-3_D的运算结果。

解 两个数用原码表示,有

$$4_D=00100 \quad -3_D=10011$$

补码:
$$4_补=00100 \quad -3_补=11101$$

运算过程如下:

$$\begin{array}{r} 0\ 0\ 1\ 0\ 0 \\ +\ \ \underset{\cdot}{1}\ \underset{\cdot}{1}\ \underset{\cdot}{1}\ 0\ 1 \\ \hline 0\ 0\ 0\ 0\ 1 \end{array}=1_D$$

【例 1.3.7】 假定固定 5 个二进制位表示数据,请结合补码求 4_D-9_D的运算结果。

解 两个数用原码表示,有

$$4_D=00100 \quad -9_D=11001$$

补码:
$$4_补=00100 \quad -9_补=10111$$

运算过程如下:

$$\begin{array}{r} 0\ 0\ 1\ 0\ 0 \\ +\ 1\ \underset{\cdot}{0}\ 1\ 1\ 1 \\ \hline 1\ 1\ 0\ 1\ 1 \end{array}=-11_D=-5_补$$

复习与思考

1.3.1 参考例 1.3.5 说明用 8 位有符号数表示的数据“－73”的补码为多少。

1.3.2 在 C 语言中，语句“printf("%d",～100);”执行结果为－101，为什么？

习题

1-1 填空题

1. 逻辑代数中的变量为逻辑量，每个变量的取值只有________两种可能。若用逻辑 1 表示________，逻辑 0 表示________，这种逻辑表示方法称为正逻辑。

2. 逻辑运算与算术运算之间存在着本质的区别。逻辑代数的基本运算有________________ 3 种，没有直接的________。当数字电路用于解决算术运算问题时，必须用________的方法来实现算术运算的功能。

3. 采用________表示二进制数的方法为规定二进制数的具体位数，超过规定位数的数据部分视为________；当数据采用固定的二进制位数表示时，一个数减去另一个数，可用这个数________另一个数来实现。

1-2 分析计算题

1. 将下列各数转换为等值的十进制数和十六进制数。

(1) $(10000001)_2$　　(2) $(01000100)_2$

(3) $(1101101)_2$　　(4) $(11.001)_2$

2. 将下列各数转换成二进制数。

(1) $(37)_{10}$　　(2) $(51)_{10}$　　(3) $(92)_{10}$　　(4) $(127)_{10}$

3. 比较下列各数，找出最大数和最小数。

(1) $(302)_8$　　(2) $(F8)_{16}$　　(3) $(1001001)_2$　　(4) $(105)_{10}$

4. 将下列各数转换成二进制数，请保持 0.001 的精度。

(1) $(0.428)_{10}$　　(2) $(0.764)_{10}$

(3) $(0.933)_{10}$　　(4) $(0.877)_{10}$

5. 将下列各数转换为等值的八进制数。

(1) $(1110000001)_{16}$　　(2) $(1101000100)_{16}$

(3) $(1101111.0001101)_{16}$　　(4) $(101111.111001)_{16}$

6. 将下列各数转换成八进制数，转换结果不低于原始数据精度。

(1) $(143.27)_{10}$　　(2) $(35.76)_{10}$

(3) $(44.43)_{10}$　　(4) $(33.67)_{10}$

7. 用二进制数算术运算完成下列各十进制数的运算。

(1) 37＋29　　(2) 49＋58　　(3) 44－37　　(4) 33－67

8. 用固定 6 位二进制数算术运算完成下列各十进制数的运算。

(1) 27+9　(2) 29+18　(3) 24−17　(4) 30−7

9. 用固定 6 位二进制数算术运算完成下列各十进制数的运算。

(1) 27+9　(2) 29+18　(3) 24−17　(4) 30−7

10. 请求下列各二进制数的补码,最高位为符号位。

(1) 1110001　(2) 01000100

(3) 101111　(4) 1010001

11. 请用 8 位二进制补码表示下列各十进制数。

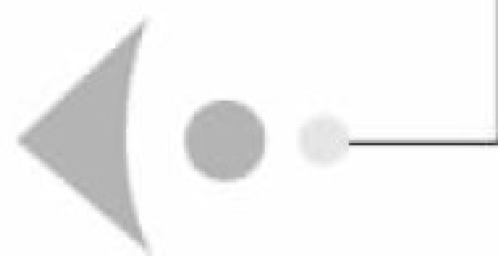
(1) 27　(2) −49　(3) −63　(4) 83

12. 请用固定 6 位二进制补码运算完成下列各十进制数的运算。

(1) 27+9　(2) 29+18　(3) 24−17　(4) 30−7



第2章



逻辑运算基础

本章要点：

本章为本书基础章，要求全部掌握。本章从3种基本逻辑运算出发，介绍常见逻辑运算特点、逻辑运算中的基本公式与常用公式；介绍逻辑函数的概念、基本运算规则及其标准形式、最简逻辑函数表达式；介绍逻辑函数表达式的公式法化简及卡诺图化简法；探讨复杂逻辑函数化简方法及逻辑函数的表达式变换。

2.1 常见逻辑运算

逻辑运算由基本逻辑运算及其组合构成，理解数字逻辑运算首先应理解基本逻辑运算。

2.1.1 3种基本逻辑运算



逻辑代数的基本运算有与、或、非3种。

1. 与逻辑运算

定义：只有当决定事件发生的所有条件都成立时，这件事件才会发生，这种因果关系称为与逻辑运算，简称与运算。

与运算用“·”号表示，逻辑符号如表2.1.1所示(为方便读者学习，基本逻辑运算及导出逻辑运算的逻辑符号及表达式汇总见表2.1.1，下同)，代数式如下：

$$Y = A \cdot B \tag{2.1.1}$$

式中，在不会导致混淆的前提下，“·”号也可以不写出。

表 2.1.1　常见运算的逻辑符号

逻辑运算	表达式	国标符号	其他符号
或运算	$Y=A+B$	≥1	+
与运算	$Y=A\cdot B$	&	
非运算	$Y=\overline{A}$	1	
与非运算	$Y=\overline{AB}$	&	
或非运算	$Y=\overline{A+B}$	≥1	+
异或运算	$Y=A\oplus B$	=1	⊕
同或运算	$Y=A\odot B$	=	⊙
与或非运算	$Y=\overline{AB+CD}$	& ≥1	+

与运算电路实例如图 2.1.1 所示。图中，只有当开关 A、B 全部闭合时，灯才能亮，具有与逻辑功能。

逻辑运算可用文字定义、电路、逻辑符号、表达式、真值表等多种方式来描述。其中，真值表是将逻辑电路中所有输入逻辑量的取值组合及其相应的输出组成的表格。

在如图 2.1.1 所示电路中，定义两个输入变量 A、B，分别对应 A、B 两个开关；定义输出变量 Y，对应灯泡；定义开关闭合、灯亮为 1，反之为 0。采用正逻辑描述本逻辑问题，具体为：开关闭合、灯亮为 1，反之为 0。根据电路的连接关系可求出真值表，如表 2.1.2 所示。

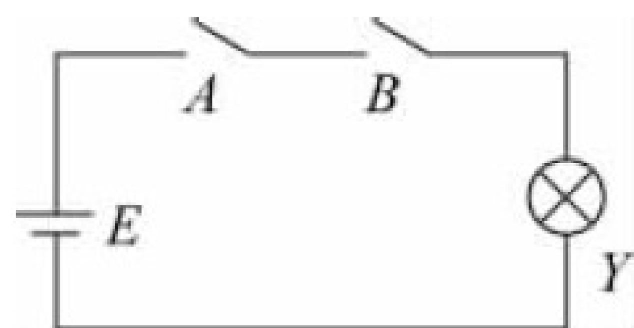


图 2.1.1　与逻辑运算[①]

表 2.1.2　与运算真值表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

① 本书中 A、B、Y、E 等符号，既表示开关、灯泡、电源等器件，也表示输入变量、输出变量等逻辑变量，且交叉出现。书中统一作斜体处理，不作区分，不影响阅读。

由表 2.1.2 可以看出，与运算的结果与数学上 2 个 1 位二进制数的乘法运算结果相同，根据这个特点，常把与逻辑称为逻辑乘。

与运算逻辑功能：0 与任何数的结果为 0。

当然，与运算并不是虚构的运算，现实生活中存在着广泛的应用实例。例如，学生宿舍开全体会议，所有同学必须到场，具有与逻辑的功能。

2. 或逻辑运算

定义：当决定事件发生的所有条件中任一个（或几个）条件成立时，这件事件就会发生，这种因果关系称为或运算，又称逻辑加。例如，联合国安理会提案表决结果与常任理事国否决票之间的关系就是或运算关系，5 个常任理事国中的 1 个投了否决票，提案即被否决。

或运算用"＋"号表示，代数式如下：

$$Y = A + B \tag{2.1.2}$$

在如图 2.1.2 所示的电路中，由开关 A 和 B 并联连接所组成的电路就是一个能实现或运算的电路。图中只要开关 A、B 中的一个闭合，灯就能亮，具有或逻辑功能。

类似地，定义两个输入变量 A、B，分别对应 A、B 两个开关；定义输出变量 Y，对应灯泡。采用正逻辑描述本逻辑问题，可求出或运算真值表，如表 2.1.3 所示。

由表 2.1.3 所示的真值表可以看出，或运算与数学上 2 个 1 位二进制数的加法运算具有很大的相似性，常把或运算称为逻辑加。

或运算逻辑功能：1 或任何数的结果为 1。

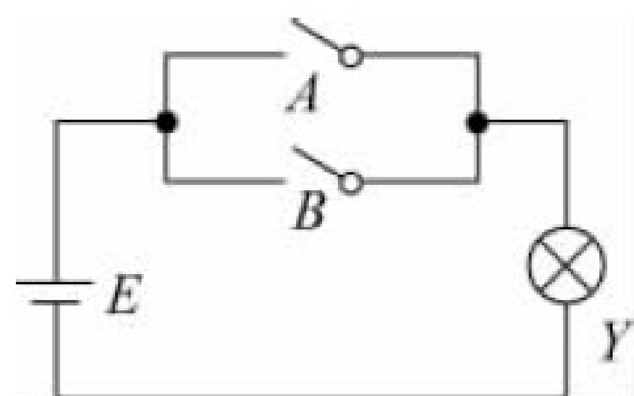


图 2.1.2　或逻辑运算

表 2.1.3　或运算真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

3. 非运算

定义：当决定事件发生的条件成立时，这件事件肯定不会发生。这种因果关系称为非运算，用"－"表示。

非运算的代数式可写为

$$Y = \overline{A} \tag{2.1.3}$$

式中："－"为运算符号。

在如图 2.1.3 所示的电路中，当开关 A 闭合时，灯亮这件事件就不会发生；反之，开关 A 断开时，灯就会亮，是一个能实现非运算的电路。

将逻辑变量 A 的取值和相应的逻辑函数值用表格表示可得到非运算的真值表，如

表 2.1.4 所示。

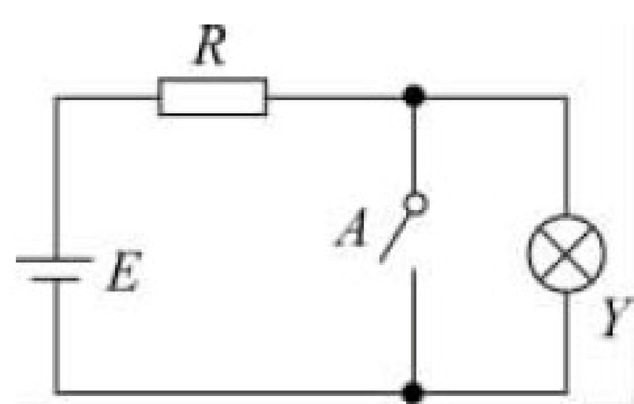


图 2.1.3　或逻辑运算

表 2.1.4　非运算真值表

A	Y
0	1
1	0

4. 3 种基本逻辑运算总结

显然，基本逻辑运算中“基本”的含义是构成逻辑运算的基础，也就是说，逻辑运算由且只能由 3 种基本逻辑运算及其组合构成。

【例 2.1.1】 下列逻辑表达式中，正确的是（　　）。

（1）1－1＝0　　　　（2）1＋1＝1

（3）1＋1＝2　　　　（4）1－0＝1

解　从数学运算角度，(1)(3)(4)均正确。

逻辑运算由且只能由 3 种基本逻辑运算及其组合构成，因此，(1)(4)中的减法运算根本不存在，(3)不符合或逻辑的定义，所以只有(2)是正确的。



2.1.2　导出逻辑运算

在逻辑代数中，除了或、与、非 3 种基本逻辑运算外，其他逻辑运算均称为导出逻辑运算。导出逻辑运算又称为复合逻辑运算。基本逻辑运算及导出逻辑运算表达式及符号如表 2.1.1 所示。

常用的导出逻辑运算如下。

1. 与非运算

与运算和非运算的组合称为与非逻辑运算，简称与非运算。表达式为

$$Y = \overline{AB} \tag{2.1.4}$$

与非运算的运算过程为先与后非，将如表 2.1.2 所示的与运算真值表取反，可以写出如表 2.1.5 所示的与非运算真值表。实现与非逻辑运算的电路称为与非门电路。

表 2.1.5　与非运算真值表

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

2. 或非运算

或运算和非运算的组合称为或非逻辑运算，简称或非运算。表达式为

$$Y=\overline{A+B} \tag{2.1.5}$$

或非运算的运算过程为先或后非，将如表2.1.3所示的或运算真值表取反，可以写出如表2.1.6所示的或非运算真值表。实现或非逻辑运算的电路称为或非门电路。

表2.1.6　或非运算真值表

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

3. 异或逻辑运算(简称异逻辑)

异或逻辑运算是只有两个输入变量的函数。只有当两个输入变量 A、B 的取值不相同时，输出才为1，否则为0。这种逻辑关系称为异或逻辑运算，简称异或运算(异逻辑)。表达式为

$$Y=A\overline{B}+\overline{A}B=A\oplus B \tag{2.1.6}$$

式中：符号⊕在或(＋)的基础上加一个圆圈，形象表示异或(不一样的或)，为异或运算的缩写符号。

根据异或运算的定义，可求出其真值表如表2.1.7所示。

表2.1.7　异或运算真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

由表2.1.7所示的真值表可以看出，异或运算不愧为“不一样的或”，实现了数学上1个二进制位的加法运算，是“真正的加”。

4. 同或逻辑运算(简称同逻辑)

同或逻辑运算是只有两个输入变量的函数。只有当两个输入变量 A、B 的取值相同时，输出才为1，否则为0。这种逻辑关系称为同或逻辑运算，简称同或运算(同逻辑)。它的表达式为

$$Y=AB+\overline{A}\overline{B}=A\odot B \tag{2.1.7}$$

式中：符号⊙为缩写符号。同逻辑的真值表如表 2.1.8 所示。

表 2.1.8　同或运算真值表

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

5. 与或非运算

或运算、与运算和非运算的组合称为与或非运算，运算顺序为先与再或，之后整体再非。根据这个运算顺序，可写出它的表达式为

$$Y=\overline{AB+CD} \tag{2.1.8}$$

请读者根据表达式自己总结与或非运算的真值表。

复习与思考

2.1.1　下列描述中不正确的是(　　)。

(1) 复合运算的特点是各运算均由基本运算组合而成

(2) 异或运算能实现 1 位二进制的加

(3) 与非运算是与和非两种基本运算的组合

(4) 以上答案至少有 1 个不正确

2.1.2　下列描述中不正确的是(　　)。

(1) 逻辑表达式 1－1＝0 当然正确

(2) 1 或任何数为 1

(3) 逻辑运算由且只能由与、或、非 3 种基本运算构成

(4) 以上答案至少有 1 个不正确

2.2　逻辑运算中的基本公式与常用公式

逻辑运算有其自身的方法和规则，掌握逻辑运算中的基本公式与常用公式是对逻辑表达式进行运算的基础。

2.2.1　逻辑运算公理



不需要加以证明，大家都公认的规律称为公理。

逻辑运算公理具体包括如下 8 个公理。

(1) 1＋1＝1＋0＝0＋1＝1(1 或任何数为 1)。

(2) 0+0=0。

(3) 0·0=0·1=1·0=0(0与任何数为0)。

(4) 1·1=1。

(5) $\overline{0}$=1。(6) $\overline{1}$=0。

(7) 若 $A \neq 1$,则 $A=0$。

(8) 若 $A \neq 0$,则 $A=1$。

从或运算、与运算、非运算的定义中,很容易理解这些公理。

2.2.2　基本公式

为方便读者,表2.2.1给出了逻辑运算的基本公式,这些公式也称布尔恒等式。

表2.2.1　逻辑代数的基本公式

序　号	公　　式	序　号	公　　式
1	$1+A=1$	9	$0 \cdot A=0, 1 \cdot A=A$
2	$0+A=A$	10	$\overline{\overline{A}}=A$(还原律)
3	$A+B=B+A$(交换律)	11	$A \cdot B=B \cdot A$(交换律)
4	$(A+B)+C=A+(B+C)$(结合律)	12	$(A \cdot B) \cdot C=A \cdot (B \cdot C)$(结合律)
5	$A \cdot \overline{A}=0$(互补律)	13	$A+\overline{A}=1$(互补律)
6	$A+A=A$(重叠律)	14	$A \cdot A=A$(重叠律)
7	$A+B \cdot C=(A+B)(A+C)$(分配律)	15	$A(B+C)=AB+AC$　(分配律)
8	$\overline{AB}=\overline{A}+\overline{B}$(反演律)	16	$\overline{A+B}=\overline{A}\,\overline{B}$(反演律)

表2.2.1中的基本公式,大多数均比较好理解。

式(1)、式(2)、式(9)为变量与常量的运算规则;式(10)称为还原律;式(5)、式(13)为变量与其反变量的运算规则,也称互补律;式(6)、式(14)为同一变量的运算规则,也称重叠律。可从逻辑量只有0、1两个值出发理解表2.2.1中的公式。

式(3)、式(11)为交换律,式(4)、式(12)为结合律,与数学运算规律吻合,均比较好理解。

式(7)、式(15)为分配律。其中式(15)与数学运算规律吻合,较好理解;可通过例2.2.1理解式(7)。

【例2.2.1】　请证明等式 $A+BC=(A+B)(A+C)$ 成立。

解　方法1,利用基本公式证明,推理过程如下:

$$(A+B)(A+C)=AA+AC+AB+BC$$

由式(14),有

$$=A+A(C+B)+BC$$

由式(9),有

$$=A[1+(C+B)]+BC$$

$$=A+BC$$

解 方法2,真值表是逻辑表达式逻辑功能的完整描述,也是唯一描述,因此,如果等式成立,则等式左、右两边的逻辑表达式逻辑功能相同,其对应的真值表也必然相同。

将 A、B、C 所有可能的取值组合逐一代入等式的两边,算出相应的结果,可得到表2.2.2所示的真值表。等式两边对应的真值表相同,所以等式成立。

表 2.2.2 $A+BC$、$(A+B)(A+C)$ 的真值表

A	B	C	BC	$A+BC$	$A+B$	$A+C$	$(A+B)(A+C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

表2.2.1中的式(8)、式(16)是著名的德·摩根定理,也称反演律。由式(8)可以看出,该表达式将与非复合运算简化为非、或两个基本运算,常用于将一个复杂的表达式变换为由基本运算构成的表达式,应用十分广泛。式(8)左边为与非组合,化为基本形式后,只能是与、或、非3种的组合,速记诀窍为"不是与",即"非、或的组合"。类似地,可利用"不是或"这个速记诀窍记忆式(16)。

2.2.3 其他常用公式

表2.2.3列出了几个常用公式。这些公式可利用基本公式导出。

表 2.2.3 若干常用公式(序号接表 2.2.1)

序号	公式
17	$A+AB=A$
18	$A+\overline{A}B=A+B$
19	$A(A+B)=A$
20	$AB+\overline{A}C=AB+\overline{A}C+BC$
21	$(A+B)(\overline{A}+C)=(A+B)(\overline{A}+C)(B+C)$

对表2.2.3中的式(17),有

$$A+AB=A1+AB=A(1+B)=A$$

对式(18),由表2.2.1中的式(7),有

$$A+\overline{A}B=(A+\overline{A})(A+B)=A+B$$

对式(19),有

$$\begin{aligned}A(A+B)&=AA+AB=A+AB\\&=A(1+B)=A\end{aligned}$$

式(20)为著名的添加项定理,证明如下:

$$\begin{aligned}AB+\overline{A}C+BC&=AB+\overline{A}C+(A+\overline{A})BC\\&=AB+\overline{A}C+ABC+\overline{A}BC\\&=AB(1+C)+\overline{A}C(1+B)=AB+\overline{A}C\end{aligned}$$

【例 2.2.2】 利用基本公式证明下面的等式成立：

$$\overline{\overline{AB}\;\overline{B+D}\;\overline{\bar{C}D}} + BC + \bar{A}\;\overline{\overline{B\bar{D}} + A + \overline{\bar{C}D}} = 1$$

解 （1）解题分析。表达式比较复杂，可利用反演律将与非、或非项变成基本运算的形式，当然，所有变换只能基于基本公式及其导出公式进行，等式左边的第 1 个与非表达式中具有 3 个与项，不可直接利用反演律进行运算，下面直接给出摩根定理的推论，即

$$\overline{AB} = \bar{A} + \bar{B} \rightarrow \overline{ABC} = \bar{A} + \bar{B} + \bar{C}$$

（2）将等式左边的第 1 个与非表达式进行变换并简化，即

$$\begin{aligned}\overline{\overline{AB}\;\overline{B+D}\;\overline{\bar{C}D}} &= \overline{\overline{AB}} + \overline{\overline{B+D}} + \overline{\overline{\bar{C}D}}\\ &= AB + B + D + \bar{C}D = B + D\end{aligned}$$

（3）将后面的或非、与非项进一步变换，有

$$\begin{aligned}&\overline{\overline{AB}\;\overline{B+D}\;\overline{\bar{C}D}} + BC + \bar{A}\;\overline{\overline{B\bar{D}} + A + \overline{\bar{C}D}}\\ &= B + D + BC + \bar{A}\;\overline{\overline{B\bar{D}}}\bar{A} + C + \bar{D}\\ &= B + D + BC + \bar{A}B\bar{D} + C + \bar{D}\\ &= B + D + C + \bar{D} = B + C + 1 = 1\end{aligned}$$

复习与思考

证明下面的等式：

$$A\bar{B} + B\bar{C} + C\bar{A} = \bar{A}B + \bar{B}C + \bar{C}A$$

2.3 逻辑函数基础

常用逻辑函数描述逻辑问题的逻辑关系，理解逻辑函数是运用数字电子技术解决逻辑问题的基础。

2.3.1 什么是逻辑函数

某逻辑网络的输入逻辑变量为 $A_1, A_2, \cdots, A_n$，输出逻辑变量为 F，当 $A_1, A_2, \cdots, A_n$ 的取值确定后，F 的值就被唯一确定下来，则称 F 是 $A_1, A_2, \cdots, A_n$ 的逻辑函数，记为 $F = f(A_1, A_2, \cdots, A_n)$。

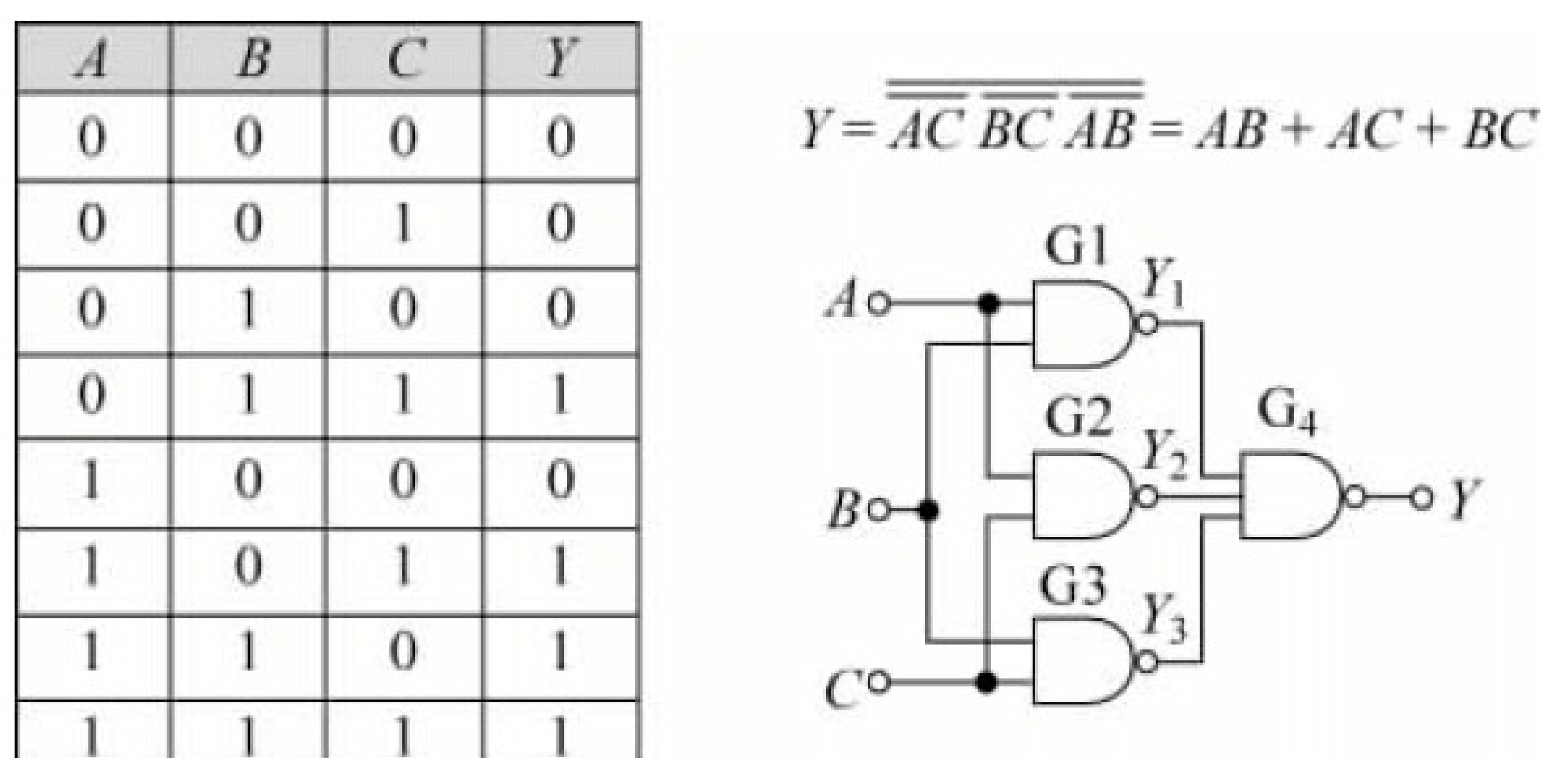
逻辑函数描述了输入逻辑变量与输出之间的一一对应关系，输出对应逻辑函数的值。以学生毕业设计答辩通过 3 人表决逻辑问题为例，3 人表决结果为输入，学生毕业设计答辩是否通过为输出，当 3 人给出表决结果后，学生毕业设计答辩是否通过结果就唯一确定，即学生毕业设计答辩是否通过结果为 3 人表决的逻辑函数。

如果两个逻辑函数具有相同的输入变量，变量所有的取值组合对应的两个函数的值均

相同，则称这两个逻辑函数相等。即两个逻辑函数相等，其真值表相同，反之亦然。

逻辑函数有真值表、表达式、逻辑电路图、卡诺图等多种表现形式。

【例 2.3.1】 请分析图 2.3.1 中真值表、电路图、表达式之间的关系。



A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

图 2.3.1　例 2.3.1 的图

解　(1) 图中的真值表、电路图、表达式具有相同的输入输出，先分析表达式 $AB+AC+BC$ 与真值表的关系。

根据或的定义，当 $AB=1$ 时，表达式 $AB+AC+BC=1$。根据与的定义，当 $AB=1$ 时，相应输入为“$A=B=1$”，对应真值表的第 6、7 行(行号对应相应输入二进制数的十进制值)。

类似地，当 $AC=1$ 时，表达式 $AB+AC+BC=1$，相应输入为“$A=C=1$”，对应真值表的第 5、7 行；当 $BC=1$ 时，表达式 $AB+AC+BC=1$，相应输入为“$B=C=1$”，对应真值表的第 3、7 行。可见，图 2.3.1 中的真值表与表达式 $AB+AC+BC$ 的真值表相同。

(2) 分析电路图与表达式的关系。写出电路对应的表达式，即

$$Y=\overline{Y_2Y_3Y_1}=\overline{\overline{AC}\;\overline{BC}\;\overline{AB}}=AB+AC+BC$$

可见，图 2.3.1 中的真值表、电路图、表达式均描述了相同的逻辑功能。其中，真值表表现方法唯一，表达式与电路图之间具有一一对应关系，可由电路图写出对应的表达式，也可由表达式画出电路图。此外，逻辑函数也可以用代数表达式来表示。但是用表达式表示逻辑函数时，不是唯一的。

例如，$Y_1=A(B+C)$，$Y_2=AB+AC$ 两个函数具有不同的表达式，但为同一函数，因为 $A(B+C)=AB+AC$。

2.3.2　逻辑抽象

当然，求解逻辑问题时，逻辑函数的各种表现形式均具有其自身的用途。真值表用于描述逻辑问题的完整功能，表达式描述逻辑问题最简洁，电路图是求解逻辑问题的最终方案。

可得出逻辑问题求解过程的一般方法如下：

逻辑问题→真值表→表达式→电路图

由现实世界的逻辑问题求出其相应逻辑代数描述的过程称为逻辑抽象。逻辑抽象的关键步骤是求出待求逻辑问题的真值表。

【例 2.3.2】 请求出两个 1 位二进制数加法的真值表。

解 (1) 定义输入及输出

用 A、B 表示两个 1 位的二进制数,用 S、C 分别表示和及进位。

(2) 求出真值表

两个 1 位二进制数加法的真值表如表 2.3.1 所示。

表 2.3.1 例 2.3.2 的真值表

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

【例 2.3.3】 1.1.1 节描述的黑白帽子逻辑问题中先问 A 所戴帽子颜色、再问 B 所戴帽子颜色,之后再问 C。A、B 均如实回答问题,试建立 C 能确定回答出帽子颜色的简化真值表。

解 用 A、B、C 分别代表 3 个人实际所戴帽子颜色(黑帽子为 1,白帽子为 0);用 Y 代表 C 能确定回答其帽子颜色,1 为能确定,0 为不能确定。A、B 不如实回答问题的输入情况不会发生,建立简化真值表时忽略 A、B 回答问题结果,可求出真值表如表 2.3.2 所示,第 2 行中×的含义将在后面解释。由表可看出,C 一定能确定其帽子颜色。如第 4 行,A 如实回答“不知道”(B、C 不是 2 个白帽子),B 确定回答“黑帽子”,C 可确定其帽子颜色为白色。本题逻辑推理过程相对复杂,可结合后面的内容逐步理解。

表 2.3.2 例 2.3.3 的真值表

A	B	C	Y
0	0	0	×
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

2.3.3 逻辑函数运算的 3 个规则



为更高效地实现逻辑函数的功能,常对逻辑函数进行变换。对逻辑函数进行适当的变换处理的原则如下:

(1) 运用逻辑运算的基本公式、常用公式进行处理。

(2) 运用逻辑运算的 3 个基本规则进行运算。

除此以外,对逻辑函数所做的任何处理均为不合逻辑的处理。

1. 代入规则

任何一个含有变量 A 的等式，如果将所有出现 A 的位置都代之以一个逻辑函数，等式仍成立，则这个规则称为代入规则。

例如在 $B(A+C)=BA+BC$ 中，将所有出现 A 的位置都代以函数 $A+D$，等式仍然成立，即

$$B[(A+D)+C]=B(A+D)+BC=BA+BD+BC$$

【例 2.3.4】 请证明等式 $A+BCD=(A+B)(A+C)(A+D)$。

解 基本公式为

$$A+BC=(A+B)(A+C)$$

依照代入规则，将 $C=CD$ 代入上式，有

$$A+BCD=(A+B)(A+CD)$$

由基本公式，有 $A+CD=(A+C)(A+D)$，代入上式，有

$$A+BCD=(A+B)(A+C)(A+D)$$

2. 反演规则

在逻辑函数 Y 的表达式中，将与（·）换成或（+），或（+）换成与（·）；再将原变量换为反变量，反变量换为原变量；并将 1 换成 0，0 换成 1；所得的新函数为原函数的反函数，记为 $\overline{Y}$。这个规则称为反演规则。

注意：利用反演规则对逻辑函数做变换时，要保持原式中的运算顺序。此外，不属于单个变量上的反号应保持不变。

【例 2.3.5】 已知 $Y=A(B+C)+CD$，求 $\overline{Y}$。

解 可利用基本公式求解，方法为先对 Y 取反，然后用摩根定理变换，化为基本运算的组合，进一步整理，可求出结果。这里直接利用反演规则求解：

$$\begin{aligned}\overline{Y}&=(\overline{A}+\overline{B}\overline{C})(\overline{C}+\overline{D})\\&=\overline{A}\overline{C}+\overline{A}\overline{D}+\overline{B}\overline{C}+\overline{B}\overline{C}\overline{D}=\overline{A}\overline{C}+\overline{A}\overline{D}+\overline{B}\overline{C}\end{aligned}$$

可见，反演规则为求解已知逻辑函数的反函数提供了方便。

【例 2.3.6】 已知 $Y=\overline{A+BC}\overline{C}D$，求 $\overline{Y}$。

解 令逻辑变量 $M=A+BC$，有 $Y=\overline{M}\overline{C}D$，根据反演规则，有

$$\overline{Y}=M+C+\overline{D}=A+BC+C+\overline{D}=A+C+\overline{D}$$

3. 对偶规则

理解对偶规则，首先应理解什么是对偶函数。

在逻辑函数 Y 的表达式中，将与（·）换成或（+），或（+）换成与（·）；1 换成 0，0 换成 1；所得的新函数为原函数的对偶函数，记为 Y^{D}。

当然，求逻辑表达式的对偶式时，要保持原式中的运算顺序，对逻辑表达式的两次对偶

即为原表达式。

若两逻辑式相等,则对偶式也相等,这就是对偶规则。对偶规则为证明两逻辑式相等提供了新的思路。当直接证明两逻辑式相等相对有难度时,可转去证明其对偶式相等,从而证明原逻辑表达式相当。

【例 2.3.7】 证明等式 $A+B\cdot C=(A+B)\cdot(A+C)$。

解　两边求对偶,有

$$A\cdot(B+C)=(A\cdot B)+(A\cdot C)$$

展开左边表达式,有

$$AB+AC=AB+AC$$

由对偶规则,可知等式成立。

此外,还可利用对偶规则记忆逻辑运算基本公式,如表 2.2.1 中式(8)(与非形式的反演律)的对偶等式为式(16)(或非形式的反演律)。

2.3.4　逻辑函数的两种标准形式



逻辑函数具有两种标准形式:最小项表达式和最大项表达式。

1. 逻辑函数的最小项表达式

理解最小项表达式首先应理解最小项的概念。

所谓逻辑函数的最小项,就是将函数的所有变量组成一与项,与项中函数的所有变量以原变量或反变量的形式仅出现一次,这种与项称为函数的最小项。

例如,三变量函数有 $2^3=8$ 个最小项,分别是 $\overline{A}\overline{B}\overline{C}$、$\overline{A}\overline{B}C$、$\overline{A}B\overline{C}$、$\overline{A}BC$、$A\overline{B}\overline{C}$、$A\overline{B}C$、$AB\overline{C}$、$ABC$。

最小项可以用符号 m_i 表示。其中的下标 i 为最小项的逻辑函数真值表对应行的二进制数值。上述 8 个最小项分别可用 m_0、m_1、m_2、m_3、m_4、m_5、m_6、m_7 表示。

根据最小项的定义,可得出最小项的主要性质如下:

(1) 任意一组变量取值,必有一个且只有一个最小项的值为 1。

(2) 同一组变量取值任意两个不同最小项的乘积为 0。

(3) 全部最小项之和为 1。

这种全部由最小项构成的与或表达式称为最小项表达式。

逻辑函数的最小项表达式是逻辑函数的标准表达式,与逻辑函数的真值表具有一一对应关系,由逻辑函数真值表求出逻辑函数 Y 最小项表达式的方法如下:

(1) 找出真值表中所有使 $Y=1$ 的行。

(2) 每行对应一个与项,其中取值为 1 的写原变量,取值为 0 的写反变量。

(3) 所有与项相或为真值表对应的最小项表达式。

【例 2.3.8】 请写出 3 人表决逻辑函数的最小项表达式。

解 (1) 先求 3 人表决逻辑函数的真值表。

所谓 3 人表决逻辑，是 A、B、C 3 个人对一提案进行表决，赞成用 1 表示，反对用 0 表示。若有两个或者两个以上的人赞成，则该提案被通过，用 1 表示；否则，该提案被否决，用 0 表示。根据此表决功能很容易写出如表 2.3.3 所示的真值表。

表 2.3.3 3 人表决逻辑函数真值表

A	B	C	Y	最小项 m_i
0	0	0	0	$\bar{A}\bar{B}\bar{C}$ m_0
0	0	1	0	$\bar{A}\bar{B}C$ m_1
0	1	0	0	$\bar{A}B\bar{C}$ m_2
0	1	1	1	$\bar{A}BC$ m_3
1	0	0	0	$A\bar{B}\bar{C}$ m_4
1	0	1	1	$A\bar{B}C$ m_5
1	1	0	1	$AB\bar{C}$ m_6
1	1	1	1	ABC m_7

为方便初学者学习，表 2.3.3 给出了真值表每行对应的最小项及其符号。

(2) 写出 3 人表决逻辑函数的最小项表达式。真值表每行对应的最小项在表 2.3.3 中已经给出，读者可对照该表理解如何由真值表写出相应的最小项。表中，有 4 行为 1，将这 4 行的最小项相或，可求出 3 人表决逻辑函数的最小项表达式如下：

$$Y = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

可用缩写式表示为

$$Y(A,B,C) = \sum_m (3,5,6,7)$$

当然，当逻辑函数是以表达式的形式给出时，也可通过表达式变换获得逻辑函数的最小项表达式。

【例 2.3.9】 求 $Y(A,B,C)=AB+AC$ 的最小项表达式。

解

$$\begin{aligned} Y(A,B,C) &= AB + AC = AB(C+\bar{C}) + A(B+\bar{B})C \\ &= ABC + AB\bar{C} + ABC + A\bar{B}C = ABC + AB\bar{C} + A\bar{B}C \\ &= \sum_m (5,6,7) \end{aligned}$$

*2. 逻辑函数的最大项表达式

逻辑函数的标准表达式，除了可以用标准与或式表示外，还可以用标准或与式表示(也称为最大项表达式)。先介绍最大项的含义。

所谓逻辑函数的最大项，就是将逻辑函数的所有变量组成一或项，或项中逻辑函数的所有变量以原变量或反变量的形式仅出现一次，这种或项称为函数的最大项。

例如，三变量逻辑函数有 $2^3=8$ 个最大项，如 $(A+B+C)$、$(A+B+\overline{C})$ 等。

根据最大项的定义，可得出最大项的主要性质如下：

(1) 任意一组变量取值，必有一个且只有一个最大项的值为 0。

(2) 同一组变量取值任意两个不同最大项的和为 1。

(3) 全部最大项之积为 0。

这种全部由最大项构成的或与表达式即为最大项表达式。

应用实践中，一般使用最小项表达式来描述逻辑函数，限于篇幅，关于逻辑函数的最大项表达式的更多阐述，请参看相关书籍。

复习与思考

指出图 2.3.2 右边哪个表达式与左边的真值表不符？

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$Y=\overline{A}BC+A\overline{B}C+AB\overline{C}+ABC$$

$$Y(A,B,C)=\sum_m(3,5,6,7)$$

$$Y=\overline{A}B\overline{C}+A\overline{B}C+AB\overline{C}+ABC$$

图 2.3.2　复习与思考的图

2.4　逻辑函数的表达式简化与变换



在数字电路的设计中，逻辑函数的表达式简化非常重要。因为逻辑函数越简单，所设计的电路就越简单。电路越简单，成本越低，稳定性越高。

2.4.1　逻辑函数的最简表达式

逻辑函数表达式具有多种形式，如例 2.3.8 中写出的 3 人表决逻辑函数 Y 为

$$Y=\overline{A}BC+A\overline{B}C+AB\overline{C}+ABC$$

该表达式为逻辑函数的标准表达式，若用电路实现，则可画出如图 2.4.1 所示的电路。从图中看出需要用 8 个门电路才能实现。如果应用前面的公式将逻辑函数 Y 化简，则有

$$\begin{aligned}Y&=\overline{A}BC+A\overline{B}C+AB\overline{C}+ABC=\overline{A}BC+A\overline{B}C+AB\overline{C}\\&\quad+ABC+ABC+ABC\\&=BC(\overline{A}+A)+AC(\overline{B}+B)+AB(\overline{C}+C)=BC+AC+AB\end{aligned}$$

根据上式可画出的电路如图 2.4.2 所示。从图中看出，只需要 4 个门就可以实现。

从上面的例子可见化简尤为重要，那么什么是最简呢？

最简与或式的标准是：与或式中，与项的数目最少，每一个与项中的变量数最少。

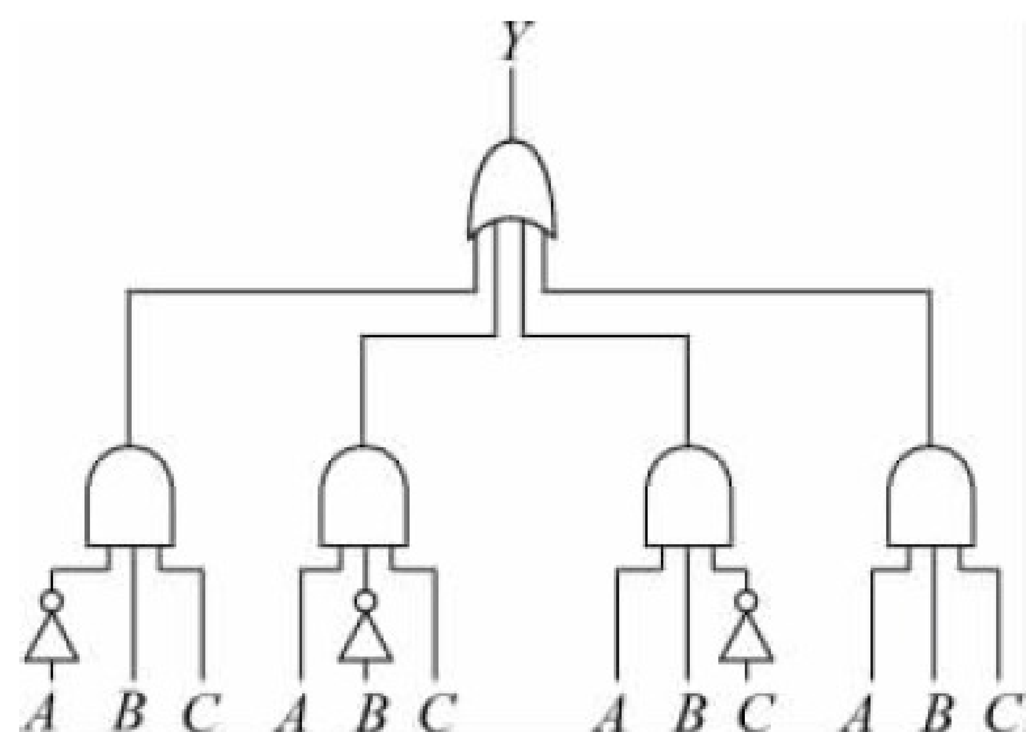


图 2.4.1 例 2.3.8 的电路图

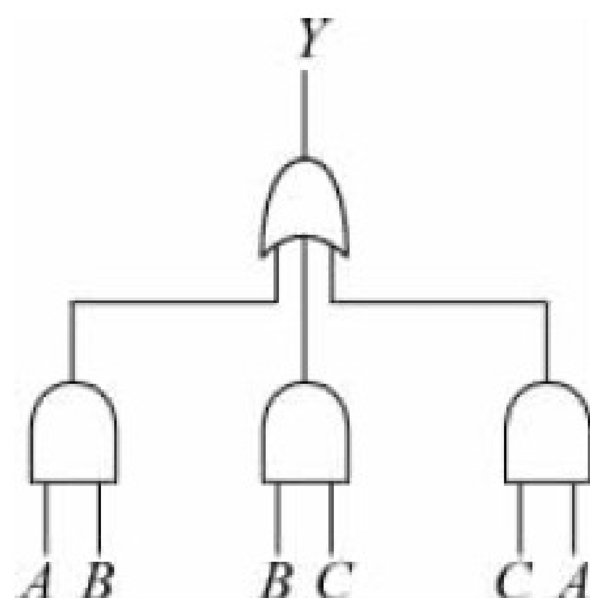


图 2.4.2 例 2.3.8 的最简图

2.4.2 逻辑函数表达式的公式法化简

所谓公式法化简，就是应用前面介绍的基本公式、常用公式、基本规则等消去逻辑函数表达式中多余的乘积项和多余的因子，以求得逻辑函数的最简与或式或者逻辑函数的最简或与式。

常用的方法如下。

1. 并项法

由表 2.2.1 中的式(13)、式(9)，有 $AB+A\overline{B}=A$。可利用该式将两项合并为一项，同时消去多余的因子 B、$\overline{B}$，从而达到化简的目的。当然，根据代入规则，A、B 均可以是任意的表达式。

2. 吸收法

利用表 2.2.3 中的式(17)($A+AB=A$)消去多余的与项 AB。类似地，A、B 均可以是任意的表达式。

3. 消项法

利用表 2.2.3 中的式(20)($AB+\overline{A}C=AB+\overline{A}C+BC$)消去多余的与项 BC。当然，A、B、C 均可以是任意的表达式。

4. 消因子法

利用表 2.2.3 中的式(18)($A+\overline{A}B=A+B$)消去与项 $\overline{A}B$ 多余的因子 $\overline{A}$。

5. 配项法

利用公式 $A=A+A$ 配项，以方便多次利用 A 结合上面的方法化简。

可通过下面的例题来理解上面的基本化简方法的运用。

【例 2.4.1】 请化简下面各逻辑函数。

(1) $Y_1=ABC+AB\overline{C}+ABD$

(2) $Y_2=\overline{\overline{AB}+\overline{C}}+\overline{\overline{AB\overline{C}}\ \overline{ABD}}$

(3) $Y_3=\overline{\overline{A\overline{C}}\ \overline{\overline{B}C}}+A\overline{B}+\overline{\overline{A\overline{B}}+\overline{C}}$

解　(1) 可用并项法等式右边将第 1、2 项合并，之后，用吸收法吸收多余的与项，即

$$Y_1=ABC+AB\overline{C}+ABD=AB+ABD=AB$$

(2) 先用反演律将表达式化为基本运算的组合，之后，利用并项法、吸收法化简，即

$$\begin{aligned}Y_2&=\overline{\overline{AB}+\overline{C}}+\overline{\overline{AB\overline{C}}\ \ \overline{ABD}}=ABC+AB\overline{C}+ABD\\&=AB+ABD=AB\end{aligned}$$

(3) 先用反演律将表达式化为基本运算的组合，之后，利用吸收法、消项法化简，即

$$\begin{aligned}Y_3&=\overline{\overline{A\overline{C}}\ \overline{\overline{B}C}}+A\overline{B}+\overline{\overline{A\overline{B}}+\overline{C}}=A\overline{C}+\overline{B}C+A\overline{B}+A\overline{B}C\\&=A\overline{C}+\overline{B}C+A\overline{B}=A\overline{C}+\overline{B}C\end{aligned}$$

【例 2.4.2】　化简以下各逻辑函数。

(1) $Y_1=AC+A\overline{B}+\overline{B+C}$

(2) $Y_2=A\overline{C}+C+\overline{A+\overline{C}}$

(3) $Y_3=A\overline{B}+BC+\overline{C+\overline{B}}$

解　(1) 先用反演律将表达式化为基本运算的组合，之后，利用消项法化简，即

$$Y_1=AC+A\overline{B}+\overline{B+C}=AC+A\overline{B}+\overline{B}\,\overline{C}=AC+\overline{B}\,\overline{C}$$

(2) 先用反演律将表达式化为基本运算的组合，之后，利用吸收法、消因子法化简，即

$$Y_2=A\overline{C}+C+\overline{A+\overline{C}}=A\overline{C}+C+\overline{A}C=A\overline{C}+C=A+C$$

(3) 先用反演律将表达式化为基本运算的组合，之后，利用并项法、消因子法化简，即

$$Y_3=A\overline{B}+BC+\overline{C+\overline{B}}=A\overline{B}+BC+B\overline{C}=A\overline{B}+B=A+B$$

*2.4.3　复杂逻辑函数化简方法探索



对于复杂的逻辑函数，一般很难直接使用 2.4.2 小节介绍的几种方法进行化简，可通过下面的例题来理解。

【例 2.4.3】　已知 $Y=AC+A\overline{B}+\overline{C}D+B\overline{D}+BD\overline{C}+AB\overline{C}$ 的最简表达式为 $Y=A+\overline{C}D+B\overline{D}$，请分析由最简式到原始式的推导过程。

解　(1) 先在 A 后添加 $AC+A\overline{B}$，即

$$Y=A+\overline{C}D+B\overline{D}=\underline{A}+(\underline{AC+A\overline{B}})+\overline{C}D+B\overline{D}$$

(2) 继续添加 $B\overline{C}$，即

$$Y=A+AC+A\overline{B}+\underline{\overline{C}D+B\overline{D}}+\underline{B\overline{C}}$$

(3) 继续添加 $A\overline{C}$，即

$$Y=A+AC+\underline{A\overline{B}}+\overline{C}D+B\overline{D}+\underline{B\overline{C}}+\underline{A\overline{C}}$$

(4) 式中包括 $A C+A\overline{C}$,可去掉 A,即

$$Y=A C+A\overline{B}+\overline{C}D+B\overline{D}+\underline{B\overline{C}}+\underline{A\overline{C}}$$

(5) 式中,$B\overline{C}$、$A\overline{C}$ 均为其余项的添加项,增加该两项的因子,即

$$Y=A C+A\overline{B}+\overline{C}D+B\overline{D}+\underline{B D\overline{C}}+\underline{A B\overline{C}}$$

根据上面的分析可知,化简该原始表达式的关键是找出 $B\overline{C}$、$A\overline{C}$ 等添加项,之后,利用前面介绍的几种方法进行化简,从而求出最简结果。

复杂逻辑函数化简的方法如下:

(1) 利用反演律将逻辑函数化为与或表达式。

(2) 拉网式搜索,找出所有可添加项。

(3) 地毯式轰炸,逐个添加看是否对化简有利。

必须指出的是,公式法化简逻辑函数需要很多技巧,并无固定的步骤,上面的解题步骤只是一种思路,并不能保证一定能求出最简单表达式。

【例 2.4.4】 化简逻辑函数 $Y=A B C+\overline{A}B\overline{C}+A B D+C D+B\overline{D}$。

解 (1) 拉网式搜索,找出所有可添加项,主要有 $B\overline{C}D$、$\overline{A}B D$、$A B$。

(2) 地毯式轰炸,添加 $B\overline{C}D$ 无直接合并项,添加 $\overline{A}B D$ 有合并项,即

$$\begin{aligned}Y&=A B C+\overline{A}B\overline{C}+\overline{A}B D+A B D+C D+B\overline{D}\\&=B D(A+\overline{A})+\overline{A}B\overline{C}+C D+A B C+B\overline{D}\\&=B D+\overline{A}B\overline{C}+C D+A B C+B\overline{D}=B(D+\overline{D})+\overline{A}B\overline{C}+C D+A B C\\&=B+\overline{A}B\overline{C}+C D+A B C=B(1+\overline{A}\overline{C}+A C)+C D=B+C D\end{aligned}$$

【例 2.4.5】 化简逻辑函数 $Y=A\overline{C}\overline{D}+B C+\overline{B}D+A\overline{B}+\overline{A}C+\overline{B}\overline{C}$。

解 (1) 拉网式搜索,找出所有可添加项,主要有 $A B\overline{D}$、$A\overline{B}\overline{C}$、$C D$、$A C$、$\overline{B}C$、$\overline{A}\overline{B}$。

(2) 地毯式轰炸,$A C$、$\overline{A}\overline{B}$ 均对化简有利,添加这两项,即

$$\begin{aligned}Y&=A\overline{C}\overline{D}+B C+A\overline{B}+A C+\overline{B}D+\overline{A}C+\overline{B}\overline{C}+\overline{A}\overline{B}\\&=A\overline{C}\overline{D}+B C+(A+\overline{A})\overline{B}+(A+\overline{A})C+\overline{B}D+\overline{B}\overline{C}\\&=A\overline{C}\overline{D}+B C+\overline{B}+C+\overline{B}D+\overline{B}\overline{C}\\&=A\overline{C}\overline{D}+C(B+1)+\overline{B}(1+D+\overline{C})=A\overline{C}\overline{D}+C+\overline{B}\end{aligned}$$

由表 2.2.3 中的式(18)($A+\overline{A}B=A+B$),得 $A\overline{C}\overline{D}+C=A\overline{D}+C$,所以

$$Y=A\overline{D}+C+\overline{B}$$

2.4.4 逻辑函数的表达式变换

在应用实践中,常使用一种器件实现逻辑函数,而前面介绍的与或表达式需要 2 种或 3 种器件才可实现。下面举例介绍将与或表达式变换为单一类型逻辑运算的方法。

【例 2.4.6】 将逻辑函数 $Y(A,B,C)=A B+A C$ 变换为全部由与非运算组成的表达式。

解 (1) 将逻辑函数两次取反,即

$$Y=A B+A C=\overline{\overline{A B+A C}}$$

（2）利用反演律将下面的与或非表达式变换为与非形式的表达式，有

$$Y=\overline{\overline{AB+AC}}=\overline{\overline{AB}\ \overline{AC}}$$

【例 2.4.7】 将逻辑函数 $Y=A\bar{D}+C+\bar{B}$ 变换为全部由与非运算组成的表达式。

解　（1）将逻辑函数两次取反，即

$$Y=A\bar{D}+C+\bar{B}=\overline{\overline{A\bar{D}+C+\bar{B}}}$$

（2）利用反演律将下面的与或非表达式变换为与非形式的表达式，有

$$Y=\overline{\overline{A\bar{D}+C+\bar{B}}}=\overline{\overline{A\bar{D}}\,\bar{C}B}$$

式中的非运算可用与非运算实现，如 $\bar{C}=\overline{C\cdot 1}$。

【例 2.4.8】 将逻辑函数 $Y(A,B,C)=AB+AC$ 变换为全部由或非运算组成的表达式。

解　（1）将逻辑函数两次取反，即

$$Y=AB+AC=\overline{\overline{AB+AC}}$$

（2）利用反演律将下面的与表达式变换为或非形式的表达式，有

$$Y=\overline{\overline{AB+AC}}=\overline{\overline{\overline{\bar{A}+\bar{B}}+\overline{\bar{A}+\bar{C}}}}$$

式中的非运算可用或非运算实现，如 $\bar{C}=\overline{C+0}$。

复习与思考

2.4.1　指出逻辑函数 $Y=A\bar{D}+C+\bar{B}$ 需要用几个门电路实现。

2.4.2　将逻辑函数 $Y=A\bar{D}+C+\bar{B}$ 变换为全部由或非运算组成的表达式并指出需要用几个门电路实现。

2.4.3　求出 3 人表决逻辑函数 $Y=\bar{A}BC+A\bar{B}C+AB\bar{C}+ABC$ 的最简表达式。

2.5　逻辑函数的卡诺图化简法

公式法化简逻辑函数需要太多的技巧，非常不直观且缺乏规范理论，因此通常使用卡诺图法化简逻辑函数。

2.5.1　逻辑函数的卡诺图表示法



先介绍最小项的逻辑相邻性的概念：若两个最小项仅有一个变量是不同的，则称它们具有逻辑相邻性。如三变量逻辑函数的两个最小项 $A\bar{B}C$、ABC，只有 B 变量不同，具有逻辑相邻性。

所谓逻辑函数的卡诺图，就是将逻辑函数的所有最小项用相应的小方格表示，并将此 2^n 个小方格排列起来，使它们在几何位置上具有相邻性，在逻辑上也相邻；反之，若逻辑相邻，几何也相邻。这种图形是由美国工程师卡诺（Karnaugh）首先提出的，故称为卡诺图。具体实现上，可将逻辑函数的真值表用方格图表示，使几何位置上相邻的最小项逻辑上也

相邻(反之亦然)。

显然,卡诺图的画法与逻辑函数的变量数相关,下面介绍常见的空白卡诺图的画法。

二变量函数有 $2^2=4$ 个最小项,其卡诺图如图 2.5.1 所示。图中小方格中的数字为该小方格相应的最小项 m_i的下标序号,类似三变量、四变量函数的卡诺图。

三变量函数有 $2^3=8$ 个最小项,其卡诺图如图 2.5.2 所示。其中的行向排列顺序为循环码"00→01→11→10"的排列顺序,以保证几何位置上相邻的最小项逻辑上也相邻。显然,从逻辑相邻角度,"00→…→10"也是相邻的,因此,几何位置上,卡诺图的首尾视为相邻。

A \ B	0	1
0	0	**1**
1	2	3

B \ A	0	1
0	0	2
1	**1**	3

图 2.5.1　二变量卡诺图

四变量函数有 $2^4=16$ 个最小项,其卡诺图如图 2.5.3 所示。其中的行、列排列顺序均为循环码的排列顺序。同理,几何位置上,四变量卡诺图的首尾也视为相邻。

A \ BC	00	01	11	10
0	0	1	3	2
1	4	5	7	6

C \ AB	00	01	11	10
0	0	2	6	4
1	1	3	7	5

图 2.5.2　三变量卡诺图

AB \ CD	00	01	11	10
00	0	1	3	2
01	**4**	**5**	**7**	6
11	**12**	**13**	**15**	14
10	8	9	11	10

CD \ AB	00	01	11	10
00	0	4	12	8
01	**1**	**5**	**13**	9
11	**3**	**7**	**15**	11
10	2	6	14	10

图 2.5.3　四变量卡诺图

五变量以上函数的卡诺图较为复杂,故五变量以上函数的化简先用公式法化简,待化简到四变量以下时,再用卡诺图化简。

下面介绍画出逻辑函数卡诺图的方法。

若已知的逻辑函数 Y 是用真值表的形式给出的,则将真值表中最小项的值"0"或"1"对号填入卡诺图中。例如,已知的逻辑函数 Y 的真值表如表 2.5.1 所示,则所对应的卡诺图

表 2.5.1　真值表实例

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

如图 2.5.4(a)(每个小方格中右下角的数字为该小方格相应的最小项 m_i 的下标序号,下同)所示。简单起见,在填"0"的小方格中,"0"可以不填进去。即在卡诺图中,未填"1"的小方格就意味着填的是"0",如图 2.5.4(b)所示。

A \ BC	00	01	11	10
0	0_{0}	1_{1}	1_{3}	1_{2}
1	1_{4}	1_{5}	0_{7}	1_{6}

(a)

A \ BC	00	01	11	10
0	$_{0}$	1_{1}	1_{3}	1_{2}
1	1_{4}	1_{5}	$_{7}$	1_{6}

(b)

图 2.5.4　表 2.5.1 对应的卡诺图

如果逻辑函数是以标准与或式给出,则将标准与或式中的最小项号码对号填入卡诺图中即可。

【例 2.5.1】　将函数 $Y(A,B,C,D)=\sum_{m}(0,1,2,3,4,5,6,10,11,12,13)$ 填入卡诺图。

解　本例中逻辑函数 Y 是以标准与或式给出,将式中最小项的脚标对应的卡诺图位置填入"1",如图 2.5.5 所示。

【例 2.5.2】　将函数 $Y=\bar{B}CD+B\bar{C}+\bar{A}\bar{C}D+A\bar{B}C$ 填入卡诺图。

解　方法 1,逻辑函数 Y 为非标准表达式,应将其变换为标准与或式后由标准与或式填写卡诺图,即

$$
\begin{aligned}
Y &= (A+\bar{A})\bar{B}CD+(A+\bar{A})B\bar{C}+(B+\bar{B})\bar{A}\bar{C}D+A\bar{B}C(D+\bar{D}) \\
&= A\bar{B}CD+\bar{A}\bar{B}CD+AB\bar{C}+\bar{A}B\bar{C}+B\bar{A}\bar{C}D+\bar{B}\bar{A}\bar{C}D+A\bar{B}CD+A\bar{B}C\bar{D} \\
&= A\bar{B}CD+\bar{A}\bar{B}CD+AB\bar{C}D+AB\bar{C}\bar{D}+\bar{A}B\bar{C}D+\bar{A}B\bar{C}\bar{D} \\
&\quad +B\bar{A}\bar{C}D+\bar{B}\bar{A}\bar{C}D+A\bar{B}CD+A\bar{B}C\bar{D}
\end{aligned}
$$

将上式填入卡诺图,如图 2.5.6 所示。

AB \ CD	00	01	11	10
00	1_{0}	1_{1}	1_{3}	1_{2}
01	1_{4}	1_{5}	$_{7}$	1_{6}
11	1_{12}	1_{13}	$_{15}$	$_{14}$
10	$_{8}$	$_{9}$	1_{11}	1_{10}

图 2.5.5　例 2.5.1 的图

AB \ CD	00	01	11	10
00	$_{0}$	1_{1}	1_{3}	$_{2}$
01	1_{4}	1_{5}	$_{7}$	$_{6}$
11	1_{12}	1_{13}	$_{15}$	$_{14}$
10	$_{8}$	$_{9}$	1_{11}	1_{10}

图 2.5.6　例 2.5.2 的图

解　方法 2,也可以作出函数 Y 的真值表后再由真值表填写卡诺图。函数 Y 的真值表如表 2.5.2 所示。

表 2.5.2　函数 Y 的真值表

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1

续表

A	B	C	D	Y
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

由真值表可填写卡诺图，如图 2.5.6 所示。



2.5.2 利用卡诺图合并最小项的规则

凡是具有逻辑相邻性的两个最小项之和都可以合并为一项，合并时能消去有关变量。基于这个原理，依照卡诺图的特点，可根据几何位置上的相邻性，寻找出计算最小项之和时的规律。

(1) 相邻的两个小方格(包括处于一行或列的两端)，可以合并为一项，合并时能够消去一个不同的变量。

例如，图 2.5.4(b)中，将相邻的两个“1”圈在一起，共可圈成 3 个圈，如图 2.5.7 所示。3 个圈的最小项分别为(“0”表示反变量、“1”表示原变量)

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

$A\overline{B}$　$\overline{A}C$　$B\overline{C}$

图 2.5.7　图 2.5.4(b)的圈组

$A\overline{B}\overline{C}+A\overline{B}C=A\overline{B}$(左边圈、消去 C，它在相邻两个最小项中不同，下同)

$\overline{A}\overline{B}C+\overline{A}BC=\overline{A}C$(中间圈、消去 B)

$\overline{A}B\overline{C}+AB\overline{C}=B\overline{C}$(右边圈、消去 A)

由此，可写出图 2.5.4(b)所示逻辑函数 Y 的最简与或式为

$$Y=\overline{A}C+A\overline{B}+B\overline{C}$$

(2) 相邻的 4 个小方格组成一个方块，或组成一行(列)，或处于两行(列)的末端，或处于四角，则可以合并成一项，合并时可以消去两个不同的变量。

例如，在例 2.5.1 所示的卡诺图中，将相邻的 4 个“1”圈在一起，共可圈成 4 个圈，如图 2.5.8 所示。由此可以写出，例 2.5.1 逻辑函数 Y 的最简与或式：

$$Y=\overline{A}\overline{C}+\overline{B}C+\overline{A}\overline{D}+B\overline{C}$$

(3) 相邻的 8 个小方格组成两行(或列)或组成两边的两行(或列)时可以合并成一项，合并时能够消去 3 个不同的变量。

例如，在图 2.5.9 所示的卡诺图中，将相邻的 8 个“1”圈在一起，共可圈成 2 个圈。由此可以写出图 2.5.9 所示的函数 Y 的最简与或式：

$$Y=\overline{A}+\overline{D}$$

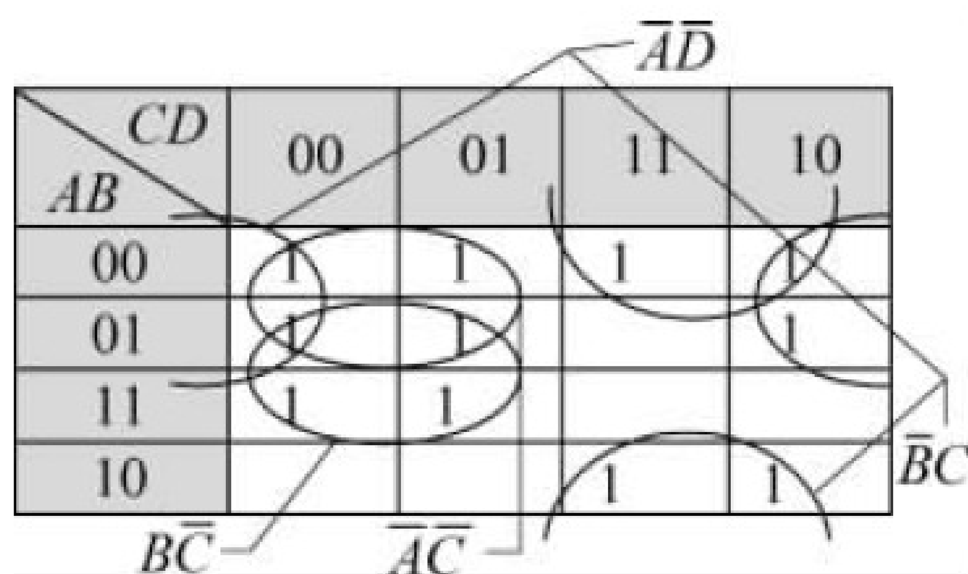


图 2.5.8 例 2.5.1 卡诺图的圈组

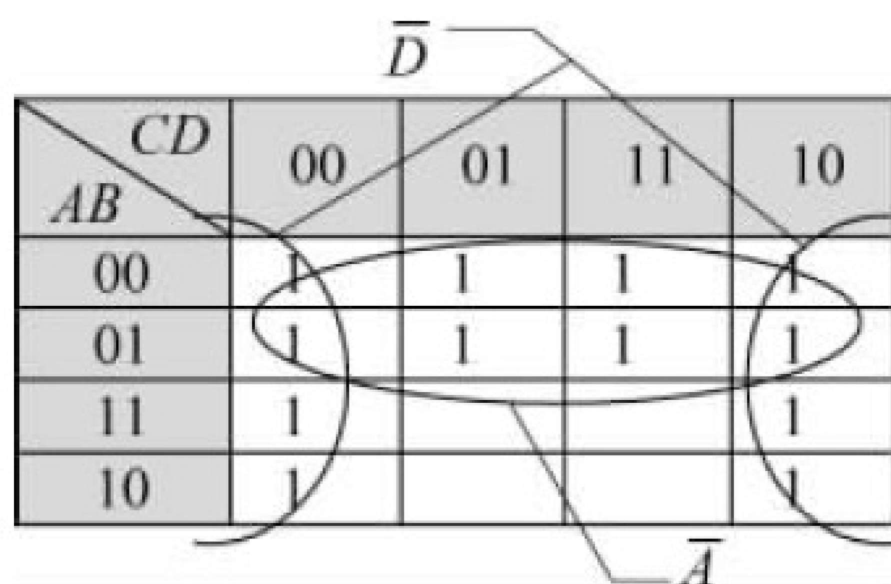


图 2.5.9 8 个小方格的圈组实例

2.5.3 利用卡诺图化简逻辑函数

由卡诺图写出逻辑函数最简与或式的方法如下。

在逻辑函数 Y 的卡诺图填“1”的小方格中：

(1) 按照 2^i($i=0,1,2,\cdots$)的相邻小方格进行最大的圈组，合并为一项，保留相同的变量，消去不同的 i 个变量。

(2) 在每一次圈组中，至少应包含一个未被圈过的小方格在内。

(3) 应将卡诺图中所有为“1”的小方格全部圈完。

(4) 将每次圈组的合并结果的与项相加就得到逻辑函数的最简与或式。

例如，在图 2.5.6 所示的卡诺图中，将相邻的 4、5、12、13 号小方格圈为一组，合并后的与项为 $B\overline{C}$；又将相邻的 1、3 号小方格圈为一组，合并后的与项为 $\overline{A}\overline{B}D$；再将 10、11 号小方格圈为一组，合并后的与项为 $A\overline{B}C$。故可写出最简与或式为

$$Y=B\overline{C}+\overline{A}\overline{B}D+A\overline{B}C$$

【例 2.5.3】 写出函数 $Y(A,B,C,D)=\sum\limits_{m}(0,1,2,4,5,8,10,11,15)$ 的最简与或式。

解 (1) 填写卡诺图。首先将函数 Y 填入卡诺图，如图 2.5.10 所示。

AB \ CD	00	01	11	10
00	1	1		1
01	1	1		
11			1	
10	1		1	1

图 2.5.10 例 2.5.3 的图 1

(2) 在填 1 的相邻小方格中进行最大的圈组。

先对 4 个相邻小方格圈组(2 项)，应包含一个未被圈过的小方格在内，具体如图 2.5.11(a)所示。再对 2 个相邻小方格圈组(1 项)，应包含一个未被圈过的小方格在内，最终圈组图如图 2.5.11(b)所示。

(3) 写出每次圈组的合并结果的与项。0、1、4、5 号小方格圈为一组，合并后的与项为 $\overline{A}\overline{C}$；0、2、8、10 号小方格圈为一组，合并后的与项为 $\overline{B}\overline{D}$；11、15 号小方格圈为一组，合并后的与项为 ACD。

(4) 将合并结果的与项相加，可得到逻辑函数的最简与或式：

AB \ CD	00	01	11	10
00	1	1		1
01	1	1		
11			1	
10	1		1	1

(a) 4个相邻小方格圈组图

AB \ CD	00	01	11	10
00	1	1		1
01	1	1		
11			1	
10	1		1	1

(b) 最终圈组图

图 2.5.11 例 2.5.3 的图 2

$$Y = \overline{A}\,\overline{C} + \overline{B}\,\overline{D} + ACD$$

【例 2.5.4】 写出函数 $Y(A,B,C,D)=\sum_m(0,2,5,6,7,9,10,14,15)$ 的最简与或式。

解 首先将函数 Y 填入卡诺图，如图 2.5.12 所示。

按照上面介绍的圈组原则进行圈组，共圈为 5 组，具体如图 2.5.13 所示。可写出最简与或式为

$$Y = C\overline{D} + BC + \overline{A}BD + \overline{A}\,\overline{B}\,\overline{D} + A\overline{B}\,\overline{C}D$$

AB \ CD	00	01	11	10
00	1			1
01		1	1	1
11			1	1
10		1		1

图 2.5.12 例 2.5.4 的图 1

AB \ CD	00	01	11	10
00	1			1
01		1	1	1
11			1	1
10		1		1

图 2.5.13 例 2.5.4 的图 2

【例 2.5.5】 写出函数 $Y=ABC+\overline{A}B\overline{C}+ABD+CD+B\overline{D}$ 的最简与或式。

解 （1）填写逻辑函数 Y 的卡诺图，如图 2.5.14 所示。

在填写卡诺图时，可先将逻辑函数变换为最小项表达式，然后在对应的小方格中填“1”即可。

当然，也可直接填写卡诺图，方法如下：

先填 ABC，在 3 个变量 A、B、C 均为 1 的小方格中填 1，对应 14、15 号小方格；再填 $\overline{A}B\overline{C}$，在 $A=0$、$B=1$、$C=0$ 的小方格中填“1”，对应 4、5 号小方格；类似地，ABD 对应 13、15 号小方格，CD 对应 3、7、11、15 号小方格，$B\overline{D}$ 对应 4、6、12、14 号小方格。

（2）画出圈组图，如图 2.5.15 所示。

AB \ CD	00	01	11	10
00			1	
01	1	1	1	1
11	1	1	1	1
10			1	

图 2.5.14 例 2.5.5 的图 1

AB \ CD	00	01	11	10
00			1	
01	1	1	1	1
11	1	1	1	1
10			1	

图 2.5.15 例 2.5.5 的图 2

（3）写出每次圈组的合并结果的与项，将合并结果的与项相加，可得到逻辑函数的最简与或式：

$$Y = B + CD$$

与例 2.4.4 结果一致。

【例 2.5.6】 化简逻辑函数 $Y=A\overline{C}\overline{D}+BC+\overline{B}D+A\overline{B}+\overline{A}C+\overline{B}\overline{C}$。

解 (1) 填写逻辑函数 Y 的卡诺图,如图 2.5.16 所示。

采用直接方法填写卡诺图,具体如下:

先填 $A\overline{C}\overline{D}$,在 $A=1$、$C=D=0$ 的小方格中填"1",对应 8、12 号小方格;再填 BC,在 $B=C=1$ 的小方格中填"1",对应 6、7、14、15 号小方格;再填 $\overline{B}D$,在 $B=0$,$D=1$ 的小方格中填"1",对应 1、3、9、11 号小方格;类似地,$A\overline{B}$ 对应 8、9、10、11 号小方格,$\overline{A}C$ 对应 2、3、6、7 号小方格,$\overline{B}\overline{C}$ 对应 0、1、8、9 号小方格。

(2) 画出圈组图,如图 2.5.17 所示。

AB \ CD	00	01	11	10
00	1	1	1	1
01			1	1
11	1		1	1
10	1	1	1	1

图 2.5.16　例 2.5.6 的图 1

AB \ CD	00	01	11	10
00	1	1	1	1
01	1		1	1
11	1		1	1
10	1	1	1	1

图 2.5.17　例 2.5.6 的图 2

(3) 写出每次圈组的合并结果的与项,将合并结果的与项相加,可得到逻辑函数的最简与或式:

$$Y=A\overline{D}+C+\overline{B}$$



2.5.4　利用无关项化简逻辑函数

在数字电路中,当分析某些逻辑函数时,发现自变量某些取值的组合根本不会出现,称为任意项。如表 2.3.2 中的第 1 行,因为只有 2 顶白帽子,因此,不会出现 3 个人均戴白帽子的情况。

"任意"二字形象地表明该取值组合输出随意,在实际应用中不会影响电路功能,因为它们根本不会出现。例如,有逻辑函数 $Y_1=A\overline{B}+C$,该逻辑函数 A、B 不可能同时为 1,即 $AB=1$ 的取值组合根本不会出现。根据逻辑函数的这个特点,有 $Y_2=A\overline{B}+C+AB=A\overline{B}+C+0=Y_1$,可见,对任意项,该取值组合输出随意。

在应用实践中还有一种情况,即某些取值组合尽管存在,但根据客观规律,不允许出现,这样的取值组合称为约束项。如 A、B 两个变量,$A=1$ 时电动机正转,$B=1$ 时电动机反转,实践中规定 A、B 不能同时为 1。这就像驾驶一辆手动挡的轿车,当前轿车处于前进运行状态,尽管倒挡挡位是可以切入的,但此时不可将变速箱切换到倒挡挡位,否则将对轿车造成伤害,应等轿车完成停止运行后再切入倒挡。

因为必须主动避免约束项出现,因此,实践中该取值组合也是不出现的,其输出也随意,不会影响电路功能。

任意项、约束项统称为无关项,在卡诺图中,用"×"(或"ϕ")表示。

可结合下面的故事理解无关项。

在西天取经的路上，唐僧因为误会将孙悟空驱逐，孙悟空要求唐僧解除他头上的金箍，唐僧说："只要我不念紧箍咒，金箍就等于没有。"显然，孙悟空头上的金箍是存在的，唐僧遵循"不念紧箍咒"的应用约束，金箍就不会发挥作用，因此，唐僧理解为"没有"。

结合上面的故事很容易理解，因为无关项在实际应用中不会出现，因此把该项当作"1"有利时，即可当作"1"；否则，可当作"0"。即在应用卡诺图化简逻辑函数时，若"×"小方格对扩大圈组范围有利，则当作"1"；否则，当作"0"。

【例 2.5.7】 写出函数 Y 的最简与或式，函数 Y 如下：

$$Y(A,B,C,D)=\sum_m(1,2,5,6,8,9)+\sum_d(10,11,12,13,14,15)$$

解 首先将逻辑函数 Y 填入卡诺图，如图 2.5.18 所示。

$\sum_d(10,11,12,13,14,15)$为任意项，因此在卡诺图中的相应小方格内填"×"。

在图 2.5.18 中，将相邻的 8、9、10、11、12、13、14、15 号小方格圈为一组，合并后的与项为 A；再将相邻的 1、5、9、13 号小方格圈为一组，合并后的与项为 $\overline{C}D$；最后将相邻的 2、6、10、14 号小方格圈为一组，合并后的与项为 $C\overline{D}$(图 2.5.19)。故可写出函数 Y 的最简与或式为

$$Y=A+\overline{C}D+C\overline{D}$$

AB \ CD	00	01	11	10
00		1		1
01		1		1
11	×	×	×	×
10	1	1	×	×

图 2.5.18 例 2.5.7 的图 1

AB \ CD	00	01	11	10
00		1		1
01		1		1
11	×	×	×	×
10	1	1	×	×

图 2.5.19 例 2.5.7 的图 2

【例 2.5.8】 写出函数 Y 的最简与或式，函数 Y 如下：

$$Y(A,B,C,D)=\sum_m(1,2,5,6,8,9)+\sum_d(10,11,12,14,15)$$

解 首先将逻辑函数 Y 填入卡诺图，如图 2.5.20 所示。

$\sum_d(10,11,12,14,15)$为任意项，因此在卡诺图中的相应小方格内填"×"。

在图 2.5.20 中，将相邻的 8、9、11、10 号 4 个小方格圈为一组，合并后的与项为 $A\overline{B}$；再将相邻的 2、6、14、10 号 4 个小方格圈为一组，合并后的与项为 $C\overline{D}$；最后将相邻的 1、3 号 2 个小方格圈为一组，合并后的与项为 $\overline{A}\overline{C}D$。剩余的 10、15 等无关项，对逻辑函数化简不利，视为 0。可画出圈组图如图 2.5.21 所示，根据圈组图，可写出函数 Y 的最简与或式为

$$Y=A\overline{B}+\overline{C}D+\overline{A}\overline{C}D$$

AB \ CD	00	01	11	10
00		1		1
01		1		1
11	×		×	×
10	1	1	×	×

图 2.5.20 例 2.5.8 的图 1

AB \ CD	00	01	11	10
00		1		1
01		1		1
11	×		×	×
10	1	1	×	×

图 2.5.21 例 2.5.8 的图 2

综上可以看出，用公式法化简逻辑函数，要求具有一定的技巧；用卡诺图化简逻辑函数，几乎不需要什么技巧，直观、易掌握。

必须指出的是，用卡诺图化简逻辑函数时，对有些逻辑函数而言，其圈组方法可以不相同。因而，其最简与或式的结果不是唯一的。例如图 2.5.7 和图 2.5.8 便具有多个最简结果。请读者自己思索其他的圈组方法。

复习与思考

2.5.1 指出选择下列图中哪个圈组是正确的圈组(见图 2.5.22)。

AB \ CD	00	01	11	10
00		1	×	1
01	×	1		1
11	×	×	×	×
10	1	1	×	×

AB \ CD	00	01	11	10
00		1	×	1
01	×	1		1
11	×	×	×	×
10	1	1	×	×

AB \ CD	00	01	11	10
00		1	×	1
01	×	1		1
11	×	×	×	×
10	1	1	×	×

AB \ CD	00	01	11	10
00		1	×	1
01	×	1		1
11	×	×	×	×
10	1	1	×	×

图 2.5.22 复习与思考 2.5.1 的图

2.5.2 指出选择下列图中哪些圈组是正确的圈组(见图 2.5.23)。

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	1		1
11	1	1		
10			1	1

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	1		1
11	1	1		
10			1	1

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	1		1
11	1	1		
10			1	1

AB \ CD	00	01	11	10
00	1	1	1	1
01	1	1		1
11	1	1		
10			1	1

图 2.5.23 复习与思考 2.5.2 的图

习题

2-1 填空题

1. 逻辑运算由且只能由________逻辑运算及其组合构成。除了或、与、非________逻辑运算外，其他的逻辑运算均称为________逻辑运算。

2. 将函数的所有变量组成一________项，________项中函数的所有变量以原变量或反变量的形式仅出现________，这种________项称为函数的最小项。在函数 Y 的真值表中，找出________的行，写出相应的最小项，然后取最小项之________，可得到逻辑函数 Y

的______________________________。

3. 将逻辑函数的所有________用相应的小方格表示，并将此 2^n 个小方格排列起来，使它们在几何位置上具有相邻性，在________上也是相邻的；反之，若________相邻，________也相邻。这样的图形称为逻辑函数的________。

4. 逻辑函数具有多种表现形式。其中，________表现方法唯一，表达式与________之间具有一一对应关系，可由________写出对应的表达式，但是用表达式表示逻辑函数时，不是________。

5. 在逻辑函数 Y 的表达式中，将与（·）换成或（＋），或（＋）换成与（·）；1 换成 0，0 换成 1；所得的新函数为原函数的________。若两逻辑式相等，则________也相等，这就是________。

2-2 分析计算题

1. 指出下列各式中，哪些是正确的逻辑表达式？（　　）

(1) $Y_1=AB+\overline{A}C$　　(2) $Y_2=\overline{AB+B\overline{C}}(A+B)$

(3) $Y_3=\overline{AB+B\overline{C}}(A-B)$　　(4) $Y_4=\overline{AB+B\overline{C}}+0$

2. 利用基本公式、常用公式法证明下列各题。

(1) $A\overline{B}+B\overline{C}+C\overline{A}=\overline{A}B+\overline{B}C+\overline{C}A$

(2) 如果 $\overline{A}\overline{B}+AB=0$，则 $\overline{AX+YB}=A\overline{X}+\overline{Y}B$

(3) 如果 $A\overline{B}+\overline{A}B=C$，则 $A\overline{C}+\overline{A}C=B$；反之亦成立

3. 指出下列各式中，哪些是四变量 A、B、C、D 的最小项。在最小项后的（　　）里填 m，其他项后填×。

(1) $AB(C+D)$（　　）　　(2) $A\overline{B}CD$（　　）　　(3) ABC（　　）

4. 在下列各逻辑函数式中，变量 A、B、C 为哪些取值时，函数值为 1？

(1) $Y_1=AB+BC+\overline{A}C$　　(2) $Y_2=\overline{A}\overline{B}+\overline{B}\overline{C}+A\overline{C}$

(3) $Y_3=A\overline{B}+\overline{A}\overline{B}\overline{C}+\overline{A}B+AB\overline{C}$　　(4) $Y_4=\overline{AB+B\overline{C}}(A+B)$

5. 已知逻辑函数 Y_1、Y_2 的真值表如题表 2.1 所示，请写出逻辑函数 Y_1、Y_2 表达式。

题表 2.1　真值表

A	B	C	Y_1	A	B	C	Y_2
0	0	0	0	0	0	0	1
0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1
0	1	1	1	0	1	1	0
1	0	0	1	1	0	0	0
1	0	1	0	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	0

6. 已知逻辑函数 Y_3、Y_4的真值表如题表 2.2 所示，请写出逻辑函数 Y_3、Y_4表达式。

题表 2.2　　真值表

A	B	C	D	Y_3	A	B	C	D	Y_4
0	0	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	0	1	1
0	0	1	0	1	0	0	1	0	0
0	0	1	1	1	0	0	1	1	0
0	1	0	0	1	0	1	0	0	1
0	1	0	1	0	0	1	0	1	0
0	1	1	0	1	0	1	1	0	1
0	1	1	1	1	0	1	1	1	1
1	0	0	0	0	1	0	0	0	0
1	0	0	1	1	1	0	0	1	1
1	0	1	0	1	1	0	1	0	1
1	0	1	1	0	1	0	1	1	0
1	1	0	0	0	1	1	0	0	1
1	1	0	1	0	1	1	0	1	1
1	1	1	0	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	0

7. 列出下列函数的真值表。

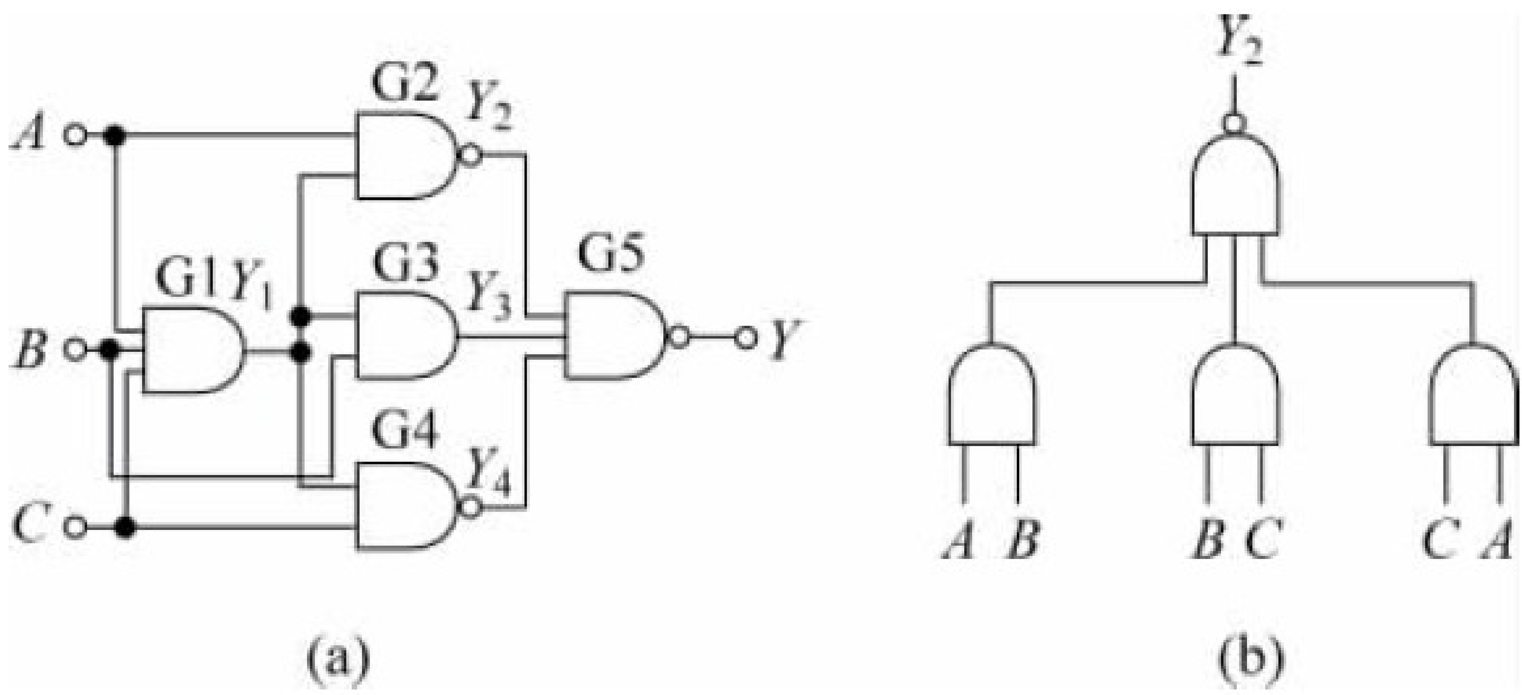
(1) $F_1(A、B、C)=AB+CA+BC$

(2) $F_2(A、B、C、D)=A\overline{C}D+\overline{A}\overline{B}+BC$

(3) $F_3(A、B、C、D)=AB+\overline{B}C+AD$

(4) $F_4(A、B、C、D)=(A+BC)\overline{C}D$

8. 写出如题图 2.1 所示各电路的逻辑函数表达式。



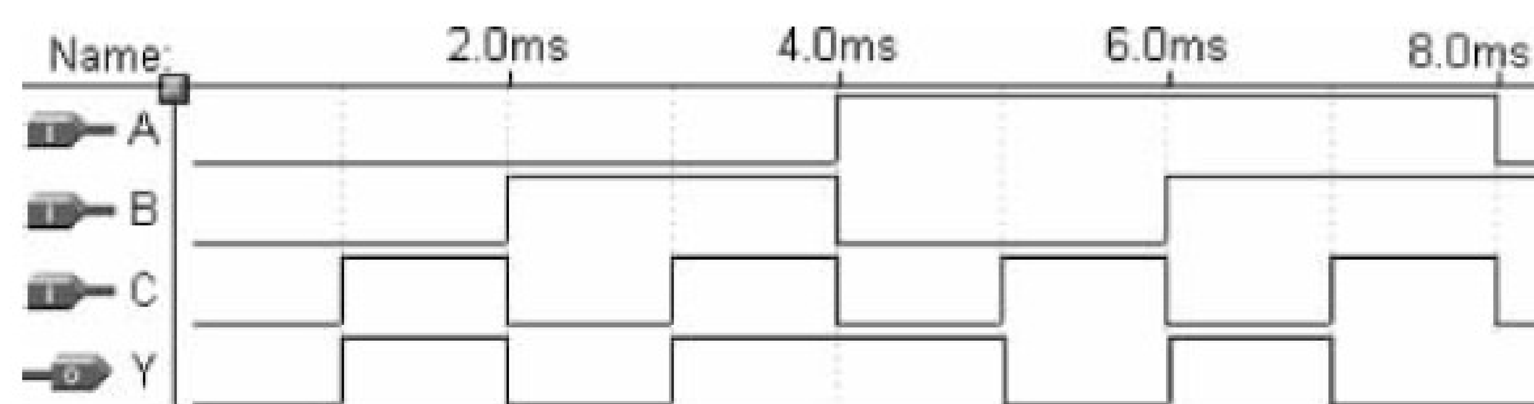
题图　2.1

9. 写出如题图 2.2 所示波形对应的逻辑函数表达式。

10. 写出下列各函数的反函数和对偶函数。

(1) $(A+\overline{B})\overline{\overline{C}+\overline{D}}$　　(2) $(\overline{A}\overline{B}+\overline{B}\overline{D})(AC+BD)$

(3) $AB+B\overline{D}+\overline{B}C+\overline{C}D$　　(4) $D[\overline{C}+(AD+B)E]$



题图 2.2

11. 将下列函数展开成最小项表达式。

(1) $F_1=AB+CA+BC$　　(2) $F_2=A\bar{C}D+\bar{A}\bar{B}+BC$

(3) $F_3=AB+\bar{B}C+AD$　　(4) $F_4=(A+BC)\bar{C}D$

12. 将下列各式化为最大项之和的形式。

(1) $F_1=AB+CA+BC$　　(2) $F_2=A\bar{C}D+\bar{A}\bar{B}+BC$

(3) $F_3=AB+\bar{B}C+AD$　　(4) $F_4=(A+BC)\bar{C}D$

13. 利用公式法化简下列函数,写出函数的最简与或式。

(1) $Y_1=AB+\bar{A}B\bar{C}$　　(2) $Y_2=A\bar{B}\bar{D}+\bar{A}\bar{B}C\bar{D}$

(3) $Y_3=A\bar{B}(A+B)$　　(4) $Y_4=AB(A+\bar{B}C)$

14. 利用公式法化简下列函数,写出函数的最简与或式。

(1) $F=A\bar{C}\bar{D}+BC+\bar{B}CD+A\bar{B}+\bar{A}C+\bar{B}\bar{C}$

(2) $F=AC+\bar{B}C+B(A\bar{C}+\bar{A}C)$

(3) $F=AC+\bar{A}BC+\bar{B}C+AB\bar{C}$

(4) $F=A\bar{B}+B\bar{C}+C(\bar{A}+D)$

(5) $F=AB+BC+AD+\bar{A}BC\bar{D}+A\bar{B}\bar{C}\bar{D}+A\bar{B}C\bar{D}$

15. 利用卡诺图化简求解第 13 题各小题。

16. 利用卡诺图化简求解第 14 题各小题。

17. 试用卡诺图化简下列函数,写出函数的最简与或式。

(1) $Y(A,B,C)=\sum_m(2,3,6,7)$

(2) $Y(A,B,C,D)=\sum_m(1,3,5,7,9,11,13,15)$

(3) $Y(A,B,C,D)=\sum_m(0,1,2,3,8,9,10,11)$

(4) $Y(A,B,C,D)=\sum_m(1,3,4,6,9,11,12,14)$

(5) $Y(A,B,C,D)=\sum_m(0,1,2,4,6,10,14,15)$

(6) $Y(A,B,C,D)=\sum_m(0,1,4,6,8,9,10,12,13,14,15)$

18. 将下列具有无关项的逻辑函数化为最简与或形式。

(1) $F=AC+\bar{A}BC+\bar{B}C+AB\bar{C}$

约束条件为 $\bar{A}B\bar{C}+A\bar{B}\bar{C}=0$

(2) $F = AB(A + \overline{B}C)$

约束条件为 $\overline{A}B\overline{C} + A\overline{B}\overline{C} = 0$

(3) $F(A,B,C,D) = \sum_m(0,1,2,3,6,8) + \sum_d(10,14)$

(4) $F(A,B,C,D) = \sum_m(0,1,2,3,6,8,11,13) + \sum_d(4,10,12,14)$

(5) $F(A,B,C,D) = \sum_m(0,1,2,6,8,9) + \sum_d(3,5,10,14)$

(6) $F(A,B,C,D) = \sum_m(1,2,3,7,8) + \sum_d(4,5,9,10,11,14)$

19. 请将下列具有逻辑函数化为与非-与非形式。

(1) $F_1(A、B、C)=AB+BC$

(2) $F_2(A、B、C、D)=A\overline{C}D+\overline{A}\overline{B}+BCD$

(3) $F_3(A、B、C、D)=AB+\overline{B}C+AD$

(4) $F_4(A、B、C、D)=(A+BC)\overline{C}D$

20. 请将下列具有逻辑函数化为或非-或非形式。

(1) $F_1(A、B、C)=AB+BC$

(2) $F_2(A、B、C、D)=A\overline{C}D+\overline{A}\overline{B}+BCD$

(3) $F_3(A、B、C、D)=AB+\overline{B}C+AD$

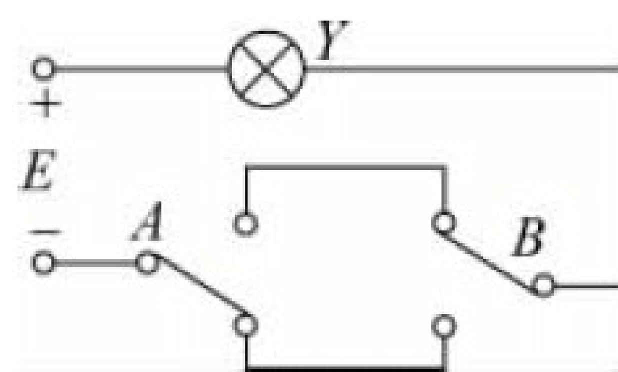
(4) $F_4(A、B、C、D)=(A+BC)\overline{C}D$

2-3　应用题

1. 三位同志各有一把锁，现将锁的打开或闭合作为逻辑输入，门的打开或关上作为逻辑输出，试说明三把锁如何构成与门、或门、与非门和或非门。

2. 男生小唐追求女生小刘，若小刘不同意，小唐便霸王硬上弓。此外，学校为所有女同学配了一瓶防狼喷雾剂。请列出小唐成功追求小刘的逻辑函数表达式。

3. 如题图 2.3 所示为一控制楼梯照明的有触点电路，在楼上、楼下各装一个单刀双掷开关 A 和 B，这样人在楼上和楼下都可以开灯和关灯。设 $Y=1$ 表示灯亮，$Y=0$ 表示灯灭；$A=1$ 表示开关向上扳，$A=0$ 表示开关向下扳，B 亦如此。试写出灯亮的逻辑表达式。

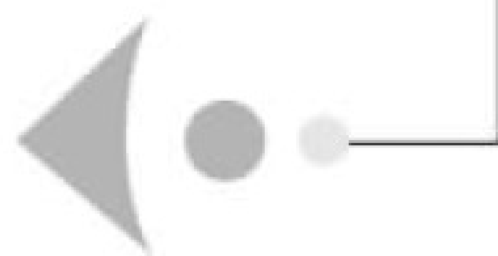


题图　2.3



第3章

逻辑运算的电路实现





本章要点：

本章为本书基础章。学习本章应重点理解晶体管的开关特性，理解 TTL 电路结构及各种输出结构电路的特点，理解 TTL 电路的输入输出特性；理解常用 CMOS 门电路构成特点，掌握 CMOS 集成逻辑门电路的特点；初步理解用 CMOS 传输门和反相器构成逻辑电路的方法及利用硬件描述语言描述常用逻辑运算的方法。

3.1 利用晶体管构成基本逻辑运算电路



逻辑运算是通过电路来实现的，实现基本逻辑运算和复合逻辑运算的单元电路通称为门电路。显然，门电路是数字电路中最基本的逻辑元件，应用极为广泛。

用门来统一描述基本逻辑运算和复合逻辑运算比较形象。首先，门具有开、关两种状态，可用于表示 0、1。此外，可以将它比作一个开关：在一定的条件下允许信号通过，称门是打开的；若条件不满足，信号就不能通过，称门是关闭的。

3.1.1 二极管的开关特性及其门电路构成特点

可利用二极管构成门电路，当然，构成门电路的器件应具有开门、关门两种状态，可在开门、关门两种状态之间相互转换，这种特性称为开关特性。

1. 二极管的开关特性

二极管具有单向导电性，即正向导通、反向截止。当二极管理想时，二极管正向导通压降等于零，相当于短路。因此，当二极管正向导通时，相当于一个接通的开关，信号允许通过，相当于开门。当二极管反向截止时，由于理想二极管反向截止电阻无穷大，反向截止电流为零，此时，二极管相当于一个断开的开关，信号不允许通过，相当于关门。

可见理想二极管的导通、截止两种状态相当于开关接通、断开两种状态，可当作开关使用。

与理想二极管相比，实际二极管正向导通压降不等于零，反向截止电流 I_0 也不为零。但当实际二极管反向截止时，由于反向截止电阻 R_0 很大，在数字电平作用下，反向截止电流 I_0 很小，可以忽略；当实际二极管正向导通时，虽然导通压降不等于零，但与数字电路高、低电平相比，比较小，不影响数字电平的高、低状态。由此可知，二极管可以当作开关元件使用。而数字电路就是利用晶体二极管的开关作用进行工作的。

当然，实际二极管从反向截止转换到正向导通、从正向导通转换到反向截止均需要时间。也就是说，二极管作为一个开关，其接通与断开均需要时间。一般情况下，二极管的接通时间很短，它对开关速度的影响很小，一般可以忽略不计。开关二极管的断开时间大约在纳秒(ns)数量级，在大多数应用场合下，均能满足要求。

2. 二极管与门

如图 3.1.1 所示为二极管与门电路($U_{CC}=10V$)，A、B 是它的两个输入端，Y 是它的输出端*。

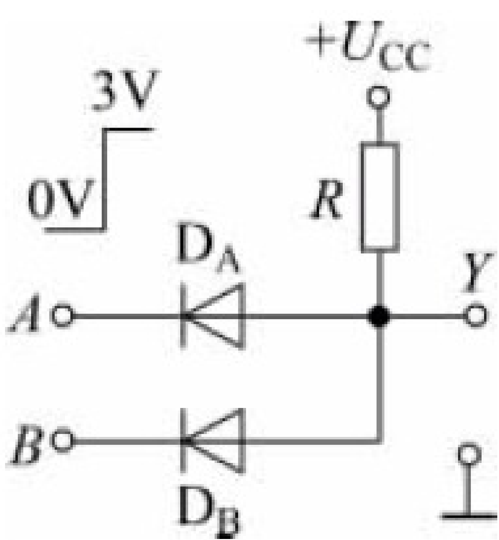


图 3.1.1　二极管与门

(1) 电压关系表。假定输入端有一个为低电平(电平可理解为电位，习惯上用 V 表示，本书中统一用 U 表示高、低电平等各种电平，其含义为该电平与零电位点的电压差，设 $U_A=0V$)，因为 $U_A=0V$，则 D_A 管优先导通，有

$$U_Y = 0.7V$$

又因为 $U_Y=0.7V$，$U_B=3V$，所以 D_B 管截止。若输入端全部为高电平($U_A=U_B=3V$)，因为 $U_A=U_B=3V$，则 D_A、D_B 管均导通，有

$$U_Y = 3.7V$$

将电路输入和输出的电压关系用表格表示，可得到表 3.1.1。

(2) 设定变量、状态赋值、列真值表。用 A、B、Y 分别表示 U_A、U_B、U_Y，用正逻辑表示，即用 0 表示低电平，用 1 表示高电平，则表 3.1.1 可转换成表 3.1.2。由表 3.1.2 看出，这是与逻辑真值表。由于图 3.1.1 所示电路是由二极管组成的，所以称为二极管与门电路。

表 3.1.1　与门电路的电压关系

U_A/V	U_B/V	U_Y/V
0	0	0.7
0	3	0.7
3	0	0.7
3	3	3.7

表 3.1.2　与门的真值表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

* 注：本书 A、B、Y 等既指输入端、输出端，又指输入信号、输出信号，且多处交叉，统一作斜体处理。

3. 二极管或门

如图 3.1.2 所示为二极管或门电路，A、B 是它的两个输入端，Y 是它的输出端。

(1) 电压关系表。假定输入端有一个为高电平(设 $U_A=3V$)，因为 $U_A=3V$，则 D_A 管导通，有

$$U_Y = 2.3V$$

又因为 $U_Y=2.3V$、$U_B=0V$，所以 D_B 管截止。

若输入端全部为高电平($U_A=U_B=3V$)，因为 $U_A=U_B=3V$，则 D_A、D_B 管均导通，有

$$U_Y = 2.3V$$

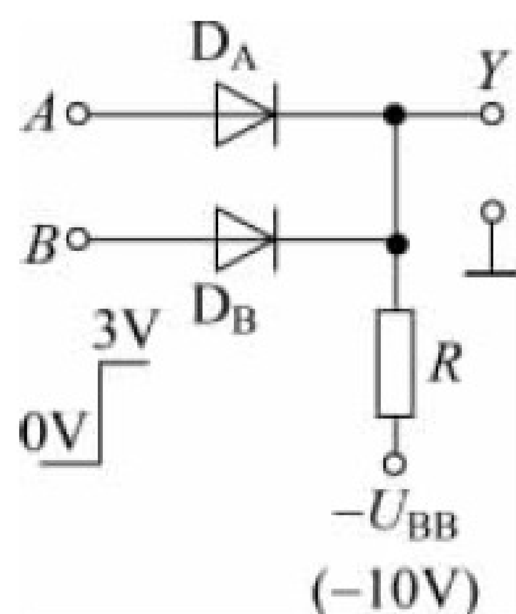


图 3.1.2　二极管或门

类似地，若输入端全部为低电平($U_A=U_B=0V$)，因电源为负值，D_A、D_B 管依旧导通，$U_Y=-0.7V$。

将电路输入和输出的电压关系用表格表示，可得到表 3.1.3。

(2) 设定变量、状态赋值、列真值表。用 A、B、Y 分别表示 U_A、U_B、U_Y，用正逻辑表示，即用 0 表示低电平，用 1 表示高电平，则表 3.1.3 可转换成表 3.1.4。由表 3.1.4 看出，这是或逻辑真值表。

表 3.1.3　或门电路的电压关系

U_A/V	U_B/V	U_Y/V
0	0	−0.7
0	3	2.3
3	0	2.3
3	3	2.3

表 3.1.4　或门的真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

读者可通过图 3.1.1 和图 3.1.2 自行总结二极管门电路的连接特点。

3.1.2　三极管的开关特性及其非门电路

1. 三极管的开关特性

三极管包括：三个区，即发射区、基区、集电区；三个极，即发射极(E 极)、基极(B 极)、集电极(C 极)；两个结，即发射结、集电结。

从工艺上看，三极管有这样的特点：发射区是高浓度掺杂区，基区很薄且杂质浓度低，集电区面积大。

三极管是一个电流控制器件，其内部结构是非常复杂的，可通过如图 3.1.3 所示仿真图来理解三极管的电流控制特性。

图 3.1.3(a)中，将三极管 B 极悬空(输出为 0mA 的电流源等同于悬空)，C、E 两极通过电阻接在 6V 电源上，测量结果显示流过的电流不到 1μA。分析电路，三极管 C、E 两极相

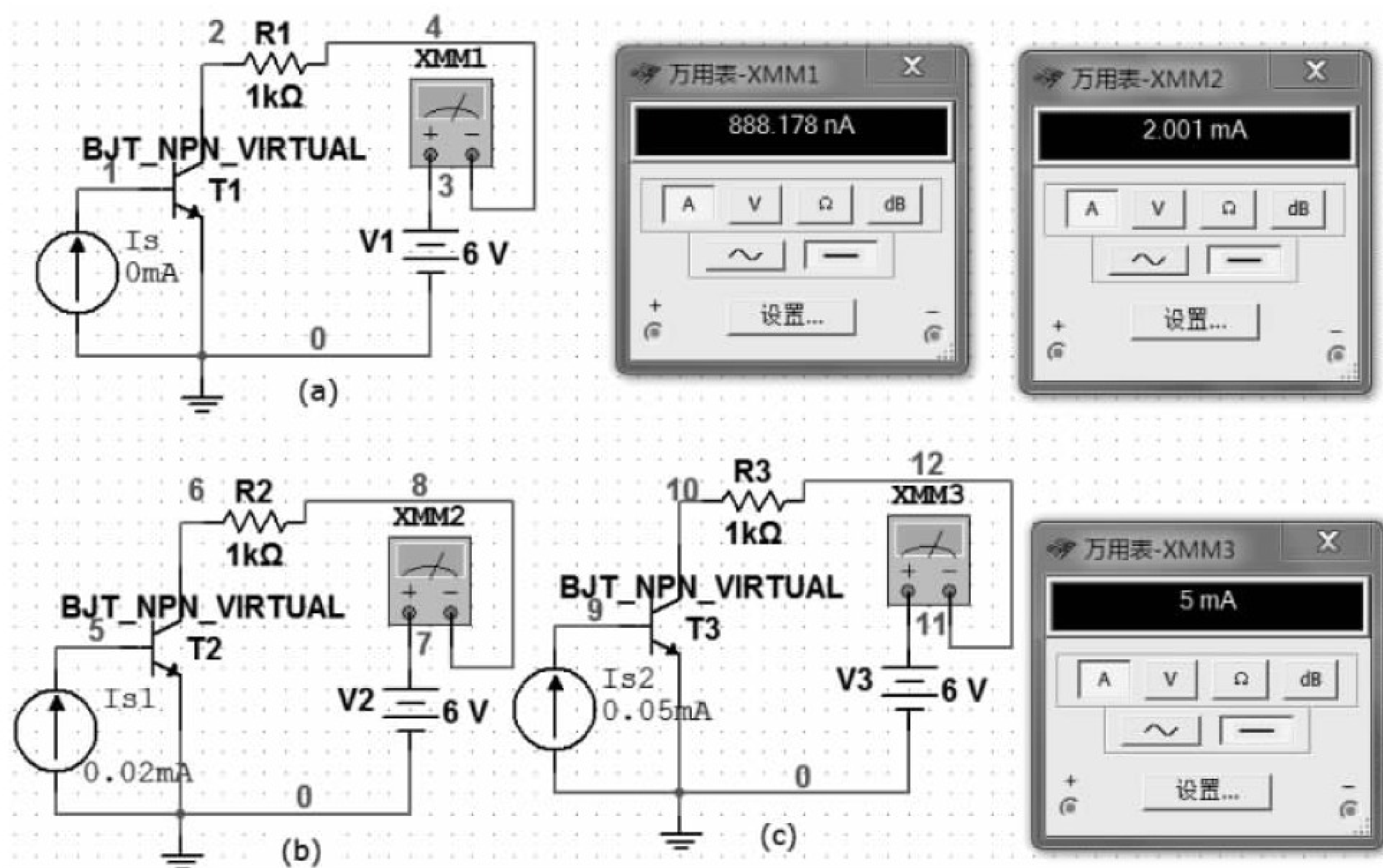


图 3.1.3 三极管电流控制特性仿真图

当于两个背靠背的二极管(图 3.1.4),由二极管的单向导电性,C、E 两极没有电流流过。

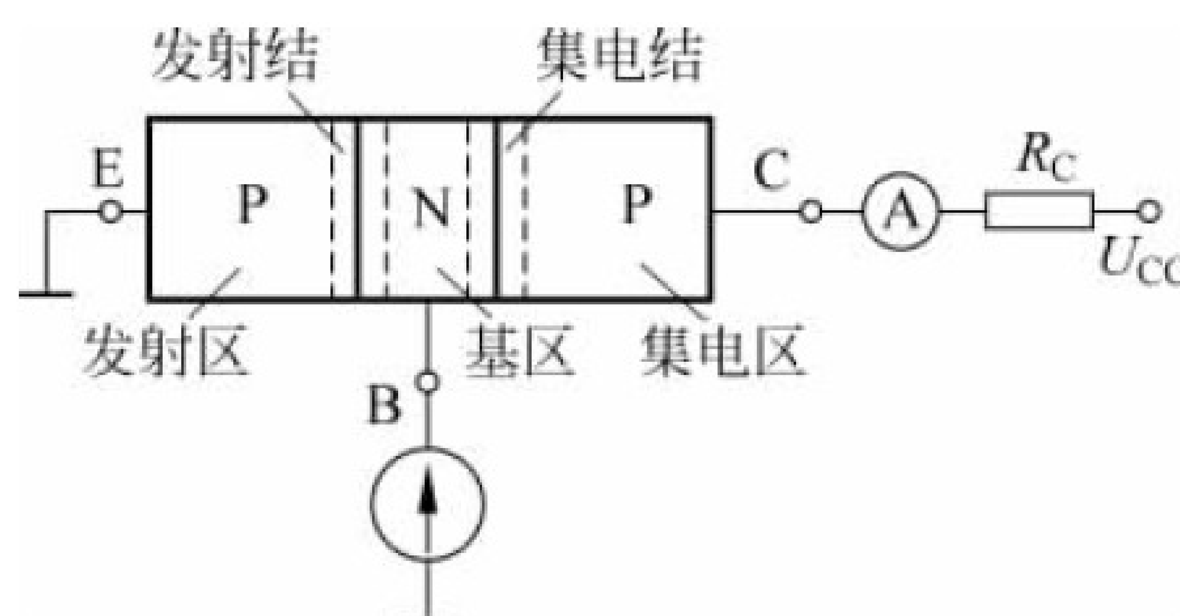


图 3.1.4 三极管电流控制特性实验电路

可见当无控制电流产生时,三极管截止,无输出电流,三极管工作在截止区。

图 3.1.3(b)图中,将三极管 B 极接 0.02mA 的电流源,C、E 两极有电流流过,大小为 2mA。

流过 B 极的电流称为基极电流,流过 C 极的电流称为集电极电流,流过 E 极的电流称为发射极电流。在三极管电路中,基极电流相当于一个控制信号,当基极电流流过基区时,改变了三极管内部导电载流子的分布,在发射结、集电结间形成了一个导电通路,C、E 两极有电流流过,这便是三极管的电流控制特性。

当控制电流大小适中时,控制电流可有效控制输出电流,输出电流与控制电流保持线性比例关系,三极管工作在放大区(图 3.1.3(b)、(c))。

显然,因电源功率等因素限制,输出电流不可能无限增长。当 I_B达到一定值以后,若基流 I_B继续增加,则集电极电流 I_C基本上不再变化,U_{CE}也基本不再变化,而且降到了很小数值。由于 I_B失去了对 I_C的控制作用,三极管也就失去了放大作用。这种现象称为饱和

现象。

可见，三极管具有截止、放大、饱和 3 种工作状态。

当发射结处于反向偏置、集电结也处于反向偏置时，三极管可靠截止。当三极管处于截止状态时，三极管 C、E 两极没有电流流过，相当于开关断开。

当发射结处于正向偏置、集电结处于反向偏置时，三极管中的电流 I_B和 I_C近似于成正比的关系，三极管处于放大工作状态。

晶体三极管工作于饱和状态时的特征为：发射结和集电结都处于正向偏置。

当三极管饱和时，C、E 两极可流过较大的电流，相当于开关闭合。

可见，三极管可以当作开关使用。而数字电路就是利用晶体三极管的开关特性进行工作的。

饱和时，B、E 间的电压记为 U_{BES}，称为饱和时的基射电压；C、E 间的电压记为 U_{CES}，称为饱和时的集射电压。对于 NPN 硅管而言，$U_{BES}=0.7V$，$U_{CES}=0.3V$。饱和时的集电极电流记为 I_{CS}，称为集电极饱和电流。

对如图 3.1.5 所示的实验电路，I_{CS}由下式决定：

$$I_{CS} = (U_{CC} - U_{CES})/ R_C \approx U_{CC}/ R_C$$

三极管刚刚出现饱和现象时的基流称为临界饱和基流，记为 I_{BS}。且有

$$I_{BS} = (U_{CC} - U_{CES})/\beta R_C$$

可总结三极管的饱和条件为

$$I_B \geqslant I_{BS} \tag{3.1.1}$$

【例 3.1.1】 在如图 3.1.6 所示电路中，$U_{CC}=12V$，$U_{BB}=12V$，$R_1=1.2k\Omega$，$R_2=10k\Omega$，$R_C=1k\Omega$，$\beta=30$，输入电压 U_I分别为 $U_{IH}=3V$，$U_{IL}=0.3V$。试问晶体三极管处于何种工作状态？

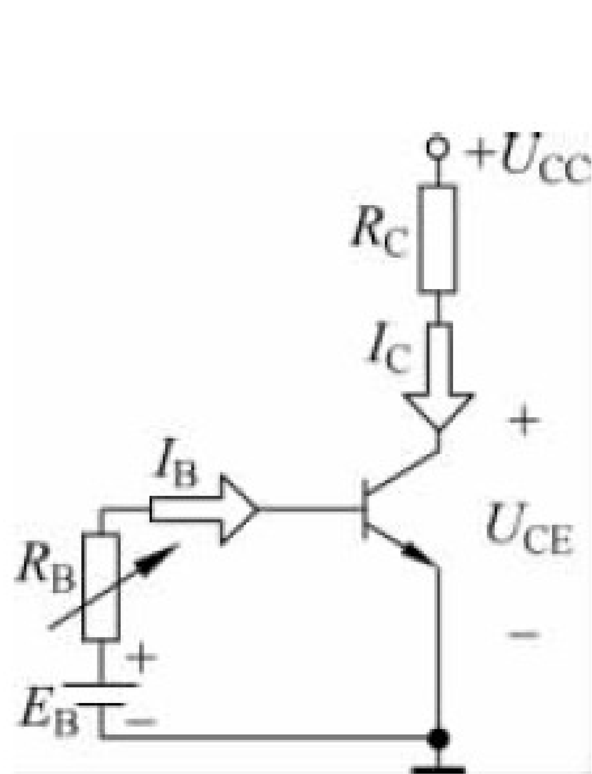


图 3.1.5 实验电路

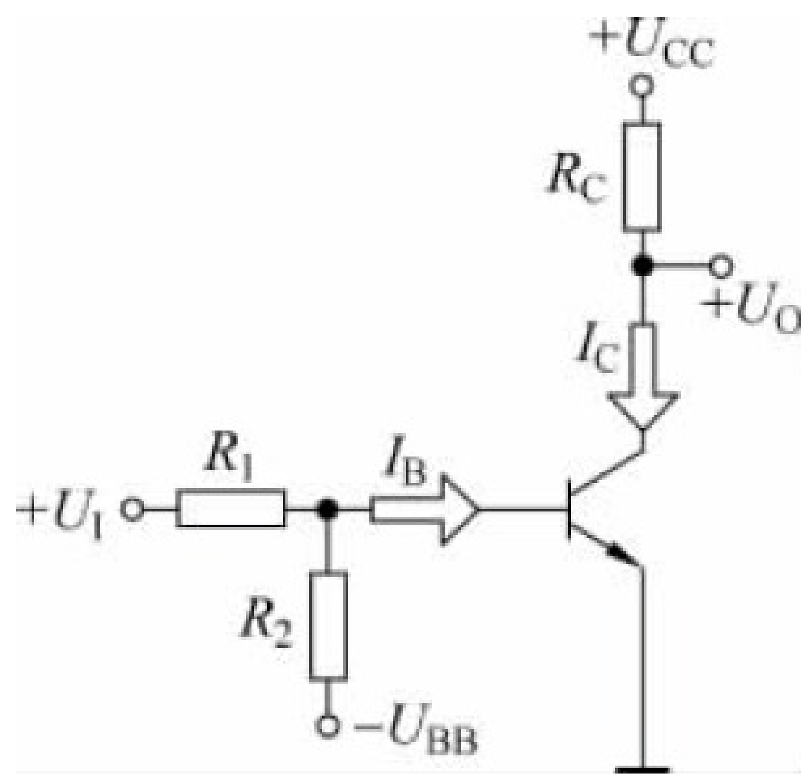


图 3.1.6 例 3.1.1 的图

解 (1) 当 $U_I=U_{IL}=0.3V$ 时，假定三极管截止。

因为三极管截止，电流 I_B可以忽略，有

$$U_{BE}= U_I - U_{R_1} = U_I - \frac{U_{BB} + U_I}{R_1 + R_2}R_1$$

$$= 0.3 - \frac{12 + 0.3}{1.2 + 10} \times 1.2 \approx -1V$$

所以，发射结处于反向偏置，晶体管截止，假定正确。

（2）当 $U_I = U_{IH} = 3V$ 时，晶体管导通，假定三极管饱和。

先求临界饱和基流：

$$I_{BS} = (U_{CC} - U_{CES})/(\beta R_C) = (12 - 0.3)/30 \approx 0.39\text{mA}$$

再求晶体管基极电流：

$$I_B = \frac{U_I - U_{BES}}{R_1} - \frac{U_{BB} + U_{BES}}{R_2} = \frac{3 - 0.7}{1.2} - \frac{12 + 0.7}{10} \approx 0.65\text{mA}$$

由于 $I_B > I_{BS}$，所以三极管工作在饱和状态，假定正确。

晶体三极管由于输入信号的作用，时而从截止状态转换到饱和状态，时而从饱和状态转换到截止状态。而在截止和饱和两种工作状态的转换中，很快经过放大状态。由于三极管内部电荷“建立”和“消散”需要一个过程，所以晶体三极管饱和与截止两种状态的相互转换也需要一定的时间才能完成。三极管从截止到饱和所需的时间称为开通时间，三极管从饱和到截止所需的时间称为关闭时间。开通时间和关闭时间总称为三极管的开关时间。它随着三极管类型的不同而有很大差别，一般为几十纳秒至几百纳秒，可以在晶体管手册中查到。三极管的开关时间限制了三极管的开关速度。开关时间越短，开关速度越快。

2. 三极管非门

如图 3.1.7 所示为由三极管组成的非门电路。A 是它的输入端，Y 是它的输出端。

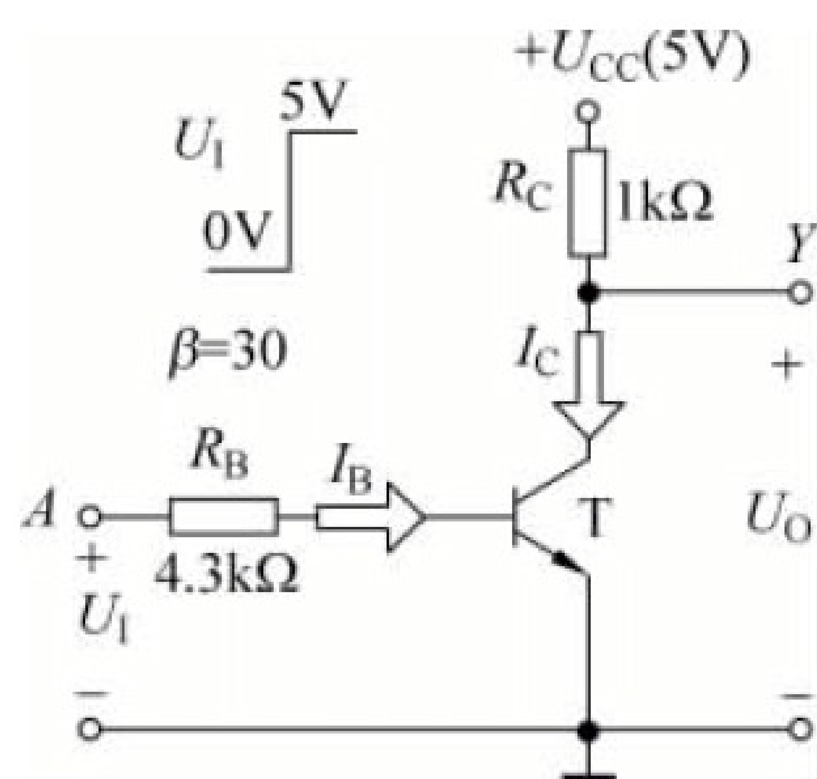


图 3.1.7　三极管非门电路

（1）电压关系表。若输入端 $U_I = U_{IH} = 5V$，则有

$$I_B = \frac{U_I - U_{BES}}{R_B} = \frac{5 - 0.7}{4.3} = 1\text{mA}$$

$$I_{BS} = (U_{CC} - U_{CES})/(\beta R_C) = (5 - 0.3)/30 \approx 0.17\text{mA}$$

由于 $I_B > I_{BS}$，所以，三极管工作在饱和状态，有

$$U_O = U_{CES} = 0.3V$$

若输入端 $U_I = U_{IL} = 0V$，有

$$U_{BE} = 0V < 0.5V$$

所以，T 管截止。输出

$$U_O \approx U_{CC} = 5V$$

将输入和输出的电压关系用表格表示，可得到表 3.1.5。

（2）设定变量、状态赋值、列真值表。用 A、Y 分别表示 U_I、U_O，用 0 表示低电平，用 1 表示高电平，则表 3.1.5 可转换成表 3.1.6。由表 3.1.6 看出，这是非逻辑真值表。同样，由于图 3.1.7 所示电路是由三极管组成的，所以称为三极管非门电路。

表 3.1.5 非门电路的电压关系

U_I/V	U_O/V
0	5
5	0.3

表 3.1.6 非门的真值表

A	Y
0	1
1	0

复习与思考

利用二极管、三极管构成一个与非门电路。

3.2 TTL 集成逻辑门电路

显然，在图 3.1.7 所示的单管非门电路中，随着负载的接入，输入、输出特性将发生显著变化。集成门电路与分立元件门电路相比，具有体积小、重量轻、可靠性高等优点。因此，在大多数领域里，集成电路已迅速取代了分立元件电路。随着集成电路制造工艺的日益完善，目前已能将数以千万计的半导体三极管集成在一片面积只有几十平方毫米的硅片上。

按照集成度（即每一片硅片中，所含有的元、器件数）的高低，可将集成电路分为小规模集成电路、中规模集成电路、大规模集成电路和超大规模集成电路。

3.2.1 TTL 与非门电路的组成特点



晶体管-晶体管逻辑（Transistor-Transistor Logic，TTL）电路是目前流行的集成系列门电路之一。

TTL 系列门电路具有标准的输入、输出特性，各种功能的 TTL 门电路组成结构大体相同，下面以如图 3.2.1 所示的典型 TTL 与非门电路为例，介绍 TTL 门电路的组成特点及其功能分析方法。

1. 电路组成与逻辑功能简述

TTL 与非门电路由以下三部分组成。

第一部分为输入级，由多发射极晶体管 T_1 和电阻 R_1 组成。

T_1 管的作用和二极管与门的作用完全相似，输入信号通过多发射极晶体管 T_1 的发射结实现了逻辑与的功能，简要分析如下：

将 T_1 管的发射结看成几个二极管，将 T_1 管的集电结看成与它们背靠背的一个二极管，如图 3.2.2 所示。该电路与如图 3.1.1 所示的二极管与门电路类似，实现了逻辑与的功能。

第二部分为中间级，由 T_2 管和电阻 R_2、R_3 组成。

中间级从 T_2 管的集电极和发射极同时输出两个相位相反的信号，作为 T_3 管和 T_4 管输出级的驱动信号，不改变电路的逻辑功能。

第三部分为输出级，由 T_3 管、D_4 管、T_4 管和电阻 R_4 组成推拉式的输出级。

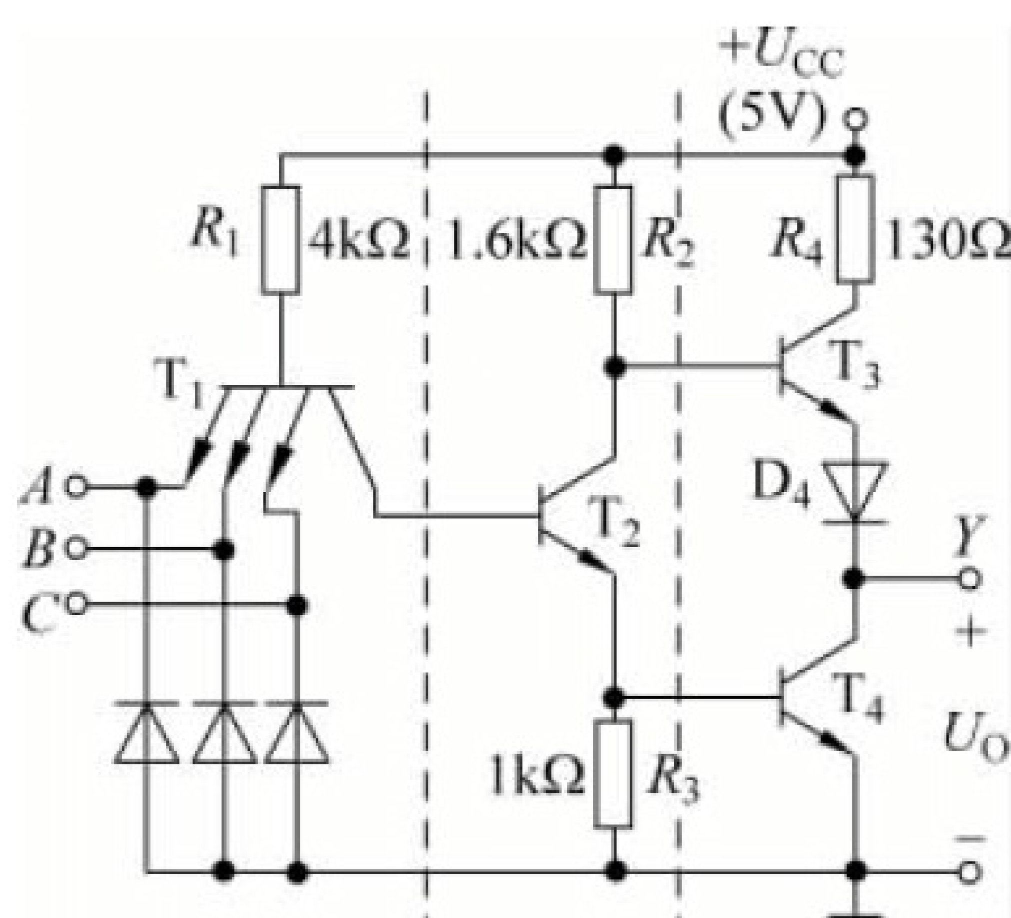


图 3.2.1 TTL与非门电路

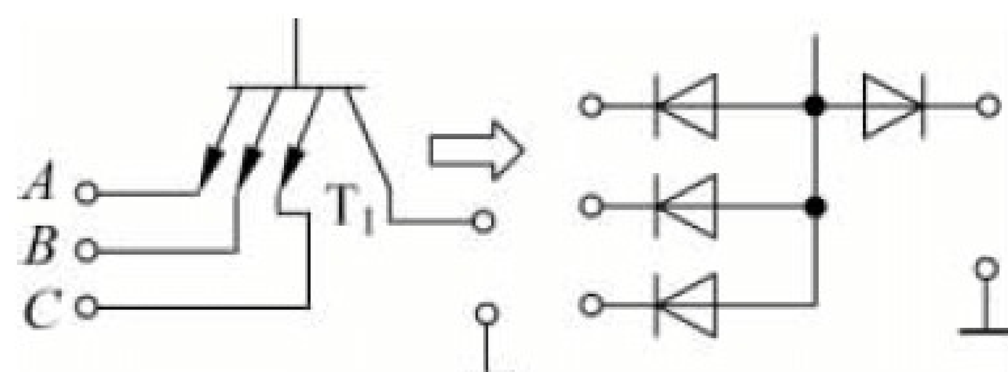


图 3.2.2 输入级电路

根据前面的分析，单个三极管实现了逻辑非运算，可见，输出级具有逻辑非的功能。

根据电路的结构可知，如图 3.2.1 所示的 TTL 电路为三输入与逻辑运算与非逻辑运算的组合运算，为三输入与非门电路。

2. 工作原理

当输入端至少有一个为低电平（设 $U_A=0.3V$，$U_B=U_C=3.6V$）时，显然，T_1管的基极 B_1与 E_{1A}发射结间处于正向偏置，电源 U_{CC}通过 R_1为 T_1管提供基极电流，其 U_{B1}为

$$U_{B1}=U_A+U_{B1E1A}=0.3+0.7=1V$$

1V 的基极电压 U_{B1}不足以给 T_1集电结、T_2发射结提供正向基极电流。所以，T_2管截止，T_4管当然也截止。因此，U_{CC}经 R_2使 T_3管、D_4管导通，则输出端的电压为

$$U_O=U_{CC}-I_{B3}R_2-U_{BE3}-U_{D4ON}\approx 5-2\times 0.7=3.6V$$

因为电流 I_{B3}为微安级，所以电压 $I_{B3}R_2$ 可以忽略不计。又由于 T_4 管截止，因此负载电流是从电源 U_{CC}经 R_4、D_4流向每个负载门的。

当输入端 $U_A=U_B=U_C=3.6V$（全部为高电平）时，T_1管瞬间导通，基极电压 $U_{T1B}=4.3V$。

4.3V 的基极电压 U_{T1B}足以给 T_1集电结、T_2发射结、T_4发射结提供正向基极电流，T_1集电结、T_2发射结、T_4发射结均导通。因为 T_1集电结、T_2发射结、T_4发射结相当于 3 个二极管，3 个二极管导通压降为 2.1V，基极电压 U_{T1B}迅速回到 2.1V。

因为输入端 $U_A=U_B=U_C=3.6V$，T_1 管的 3 个发射结均处于反向偏置，截止。电源 U_{CC}经过电阻 R_1 和 T_1 管的集电结向 T_2 管提供足够的基极电流，使 T_2 管导通并饱和，T_2管的发射结电流又给 T_4 管提供足够的基流，使 T_4 管也导通并饱和。此时，输出端的电压为

$$U_O=U_{CES4}=0.3V=U_{OL}$$

此时，由于 T_2管的集电极电位为

$$U_{C2}=U_{CES2}+U_{BES4}=0.3+0.7=1V$$

所以，T_3管、D_4管截止。由于D_4管截止，负载电流全部灌入T_4管的集电极。

将输入和输出的电压关系以表格表示，可得表3.2.1。

用A、B、C、Y分别表示U_A、U_B、U_C、U_O，用0表示低电平，用1表示高电平，则表3.2.1可转换成表3.2.2。由表3.2.2看出，这是与非逻辑的真值表。

表3.2.1 TTL与非门的电压关系（单位：V）

U_A	U_B	U_C	U_O
0.3	0.3	0.3	3.6
0.3	0.3	3.6	3.6
0.3	3.6	0.3	3.6
0.3	3.6	3.6	3.6
3.6	0.3	0.3	3.6
3.6	0.3	3.6	3.6
3.6	3.6	0.3	3.6
3.6	3.6	3.6	0.3

表3.2.2 TTL与非门的真值表

A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

可见，如图3.2.1所示电路为TTL与非门电路。即

$$Y = \overline{ABC}$$

3.2.2 TTL与非门电路的输入输出特性

常用输入电压、输入电流、输入电阻等参数描述电路的输入特性，用输出电压、输出电流、输出电阻等参数描述电路的输出特性。

当然，数字系统具有其特殊性，其输入只有1、0两种值，对应高、低两种电平值。根据前面的分析，TTL与非门电路输出为1时，有

$$U_{OH}(\text{最大}) = 3.6\text{V}$$

TTL与非门电路输出为0时，有

$$U_{OL} = 0.3\text{V}$$

1. 输入特性

在标准TTL电平作用下，对输入回路，可忽略T_2集电极、T_4集电极、R_3。取多输入极的A极为例分析，参考电路如图3.2.3所示。图中，$U_{CC}=5\text{V}$。

正常情况下，TTL电路的输入来自TTL电路的输出，为标准值，有

$$U_{IH} = U_{OH}(\text{最大}) = 3.6\text{V}$$

$$U_{IL} = U_{OL} = 0.3\text{V}$$

因为输入电平为标准值，为已知参数，所以TTL电路的输入特性主要用输入电流来描述。

分析如图3.2.3所示的输入级电路，当$U_{IL}=U_{OL}=0.3\text{V}$时，$T_1$集电结截止，有

$$I_{IL}=\frac{(0.3+0.7)-5}{R_1}=\frac{-4}{4\times10^3}=-1(\text{mA})$$

分析如图 3.2.3 所示的输入级电路，当 $U_{IH}=U_{OH}=3.6\text{V}$ 时，根据前面的分析，T_1 发射结瞬间导通后截止，输入电流为二极管反向截止电流，I_{IH}的值一般在 $40\mu\text{A}$ 以下。

当然，输入电平从低电平变化到高电平、从高电平变化到低电平均存在一个变化过程。输入电流随输入电平变化的曲线称为输入特性曲线，如图 3.2.4 所示。对输入信号的电平转换的暂态过程分析，读者可参考其他书籍。

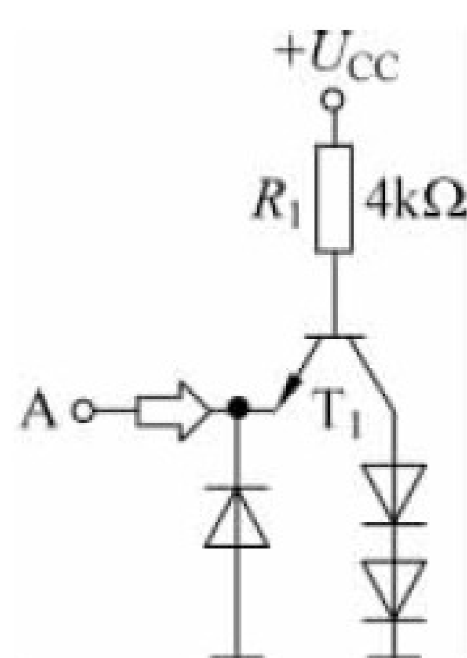


图 3.2.3 输入等效电路

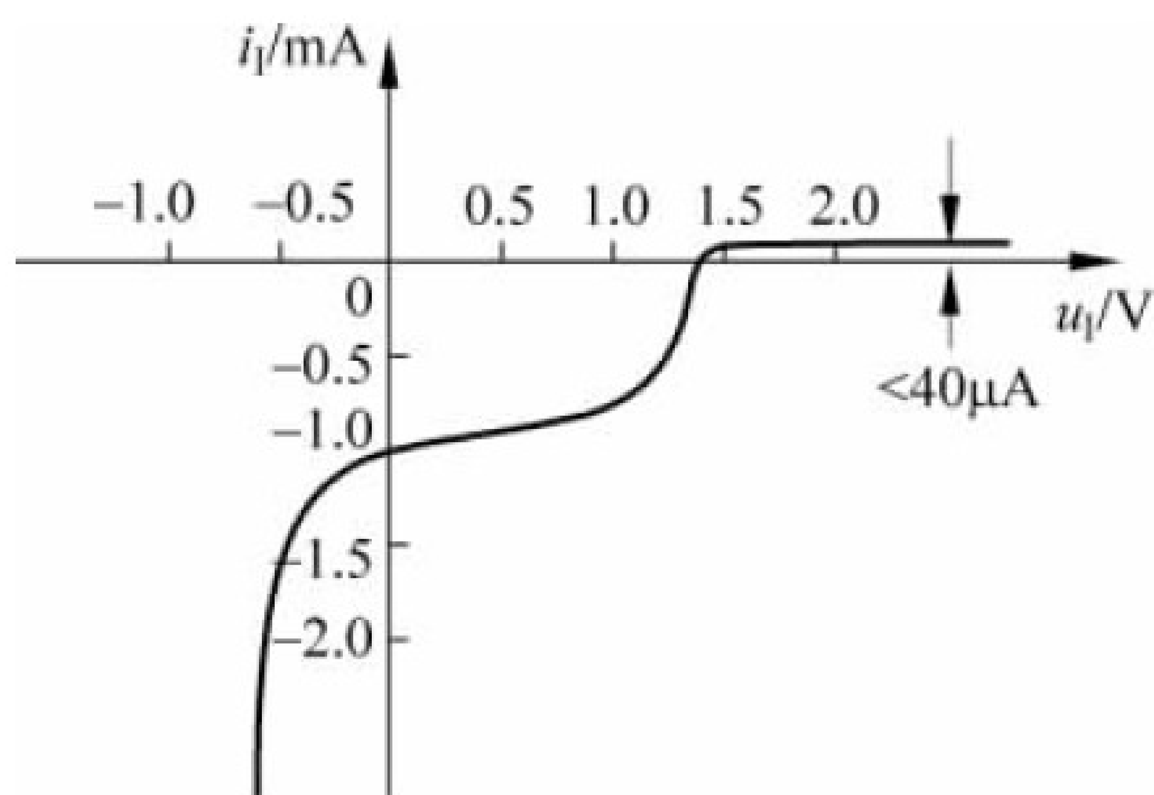


图 3.2.4 输入特性曲线

2. 高电平输出特性

在如图 3.2.1 所示 TTL 门电路中，当至少有 1 个输入为低电平时，输出为 1。对输出回路，可忽略 T_2管、T_4管，参考电路如图 3.2.5 所示。图中，$U_{CC}=5\text{V}$。

正常情况下，输出的高电平也为标准值，为已知参数，因此，TTL 电路的输出特性也主要用输出电流来描述。

分析如图 3.2.5 所示输出等效电路，当未接负载时，$U_{OH}=3.6\text{V}$，没有电流流出。当输出接负载时，U_{CC}将通过 T_3管、D_4管向外输出电流，这种负载电流称为拉电流。

当然，随着负载电流绝对值的增加，电阻 R_4上的压降也将随之增加，相应输出的电位值将减小。高电平输出电位随输出电流变化的曲线称为高电平输出特性曲线，如图 3.2.6 所示。

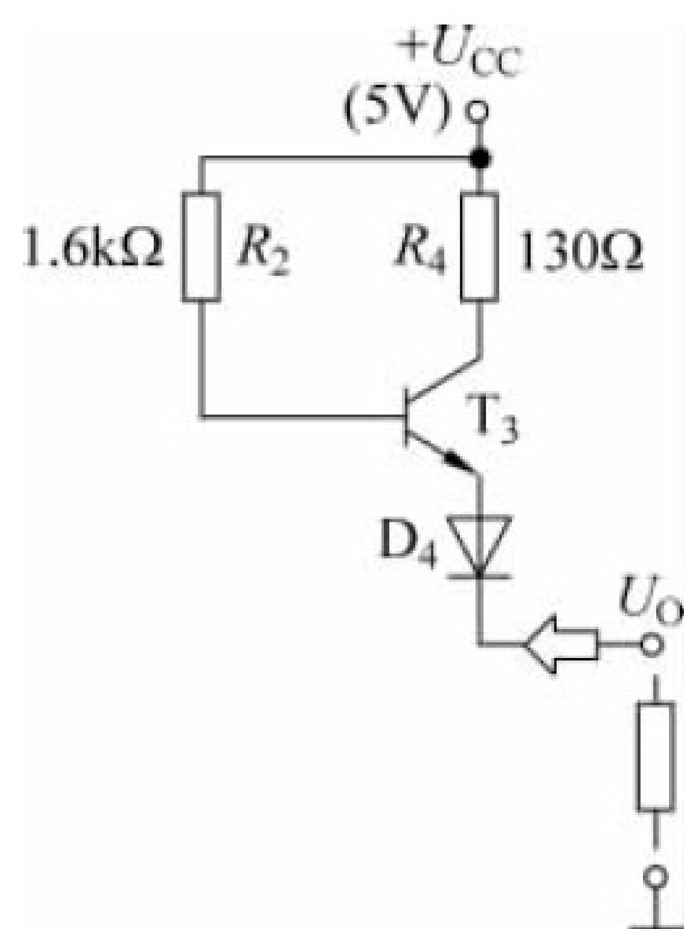


图 3.2.5 高电平输出等效电路

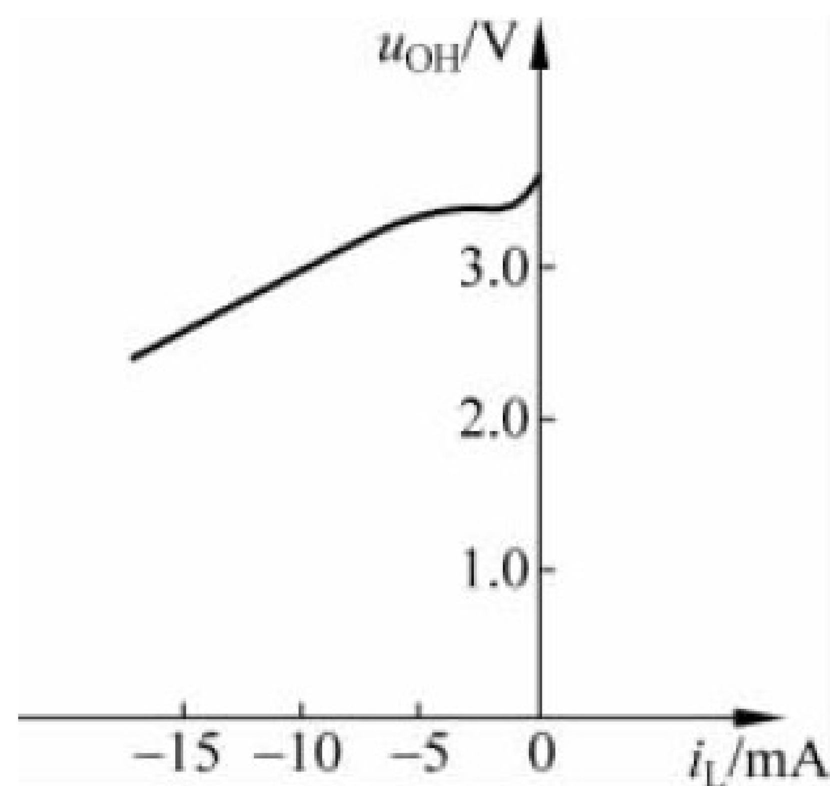


图 3.2.6 高电平输出特性曲线

从输出特性曲线可以看出，当输出电流绝对值小于 5mA 时，输出的高电平的值不会明显下降，可保证高电平的稳定。

必须指出的是，考虑到功耗的限制，TTL 集成门电路手册上给出的高电平输出电流最大值比 5mA 小得多。例如，74 系列门电路的运用条件规定，输出为高电平时，流过负载上的电流不得超过 0.4mA。

3. 低电平输出特性

在如图 3.2.1 所示 TTL 门电路中，当 3 个输入为高电平时，输出为 0。对输出回路，可忽略 T_3管、D_4管，参考电路如图 3.2.7 所示。

分析如图 3.2.7 所示输出等效电路，T_4管饱和导通，负载电流经 T_4管从电源“＋”端流向电源“－”端，负载电流灌入 T_4管，这种负载电流称为灌电流。

此外，T_4管 C-E 间的饱和导通内阻很小，通常在 10Ω 以内。因此，当负载电流增加时，相应输出的低电平电位值仅略有升高，能在较大范围内保证低电平的稳定，低电平输出电位随负载电流变化的曲线如图 3.2.8 所示。

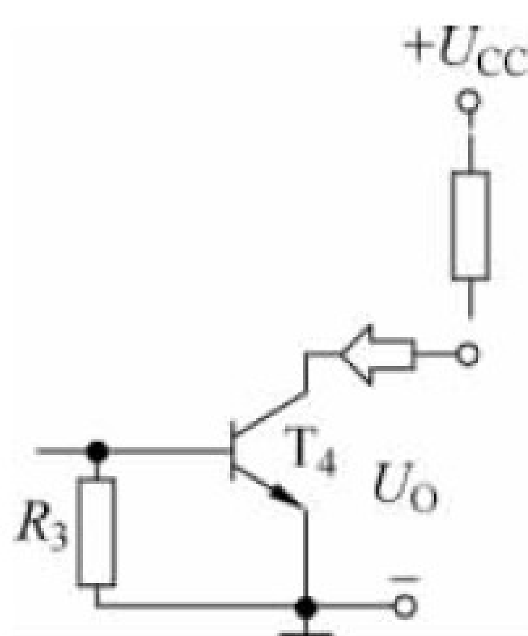


图 3.2.7 低电平输出等效电路

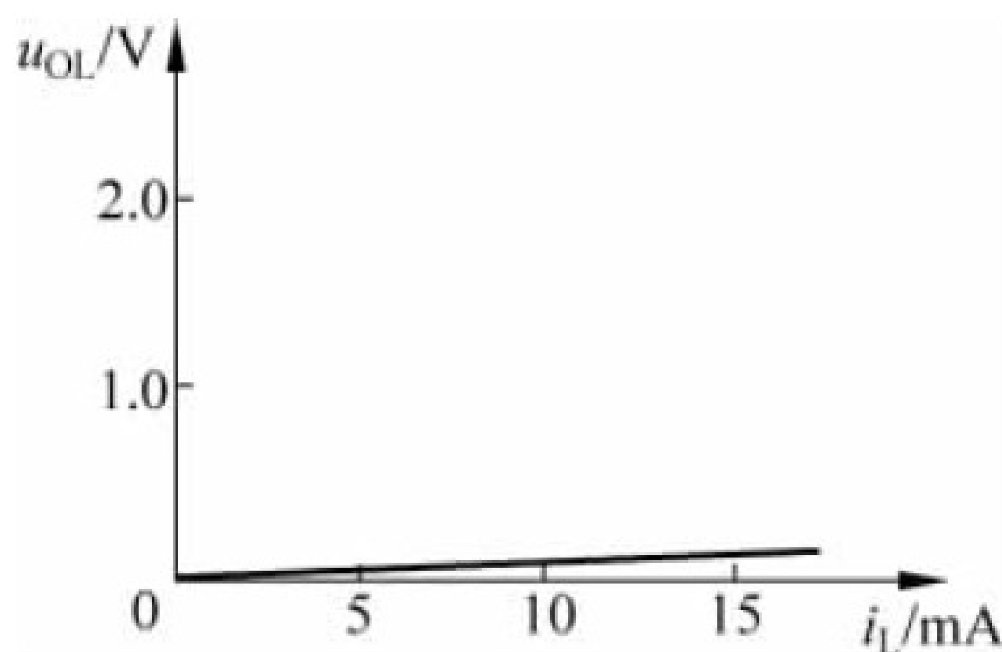


图 3.2.8 低电平输出特性曲线

4. 电压传输特性



所谓 TTL 与非门的电压传输特性，是指输出电压 U_O随输入电压 U_I的变化曲线。如图 3.2.9 所示为图 3.2.1 所示与非门电路的电压传输特性曲线。它是通过实验测出来的，即将某一个输入端的电压由零逐渐增大，而将其他输入端接在电源正极以保持恒定高电位。由图可见，曲线可分为 AB、BC、CD、DE 四段。

AB 段：在这一段里，$U_I<0.6$V 使 $U_{B1}<1.3$V，则 T_2管、T_4管截止，而 T_3管、D_4管导通，输出电压 $U_O=3.6$V，为高电平。这一段称为截止区。

BC 段：当 U_I为 0.6～1.3V 时，U_B的增加使 T_2管开始导通并进入到放大区，但 T_4管仍然截止，则 U_O随着 U_I的增加而线性地减小。这一段称为线性区。

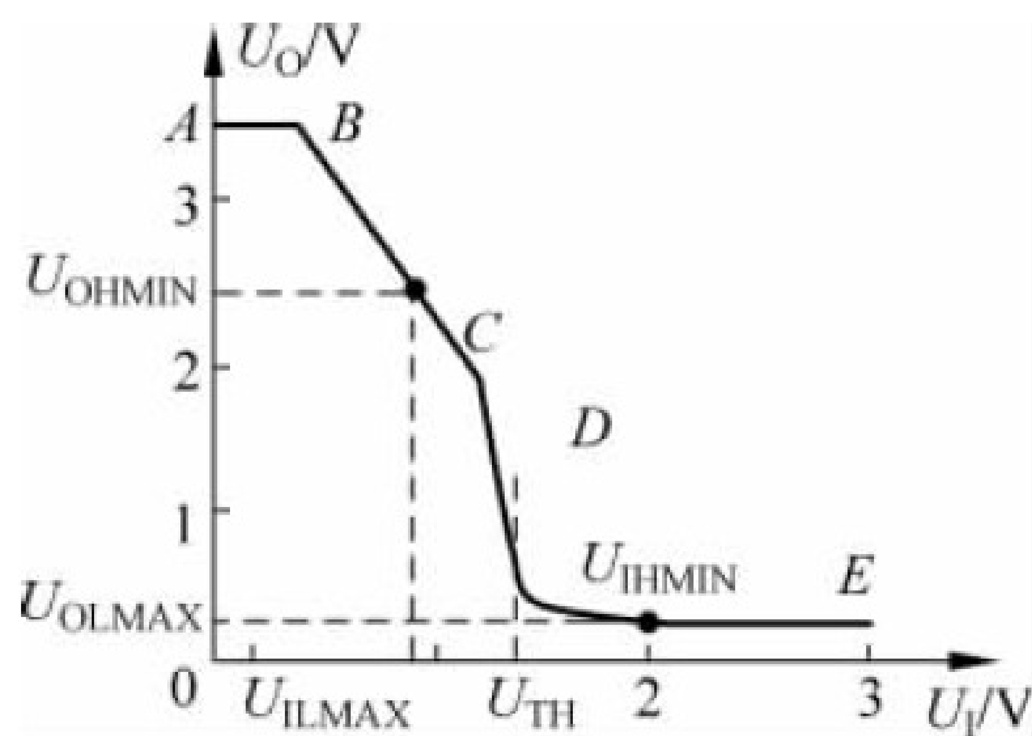


图 3.2.9 电压传输特性

CD 段：当 U_I增加到 1.4V 左右时，T_4管也开始导通，使输出电压 U_O急剧下降，这一段称为转折区。在转

折区中心点所对应的输入电压称为与非门的阈值电压或称为门槛电平，用 U_{TH} 表示。从图 3.2.9 中可知，$U_{TH}=1.4V$。

DE 段：当 $U_I>1.4V$ 时，即特性曲线经过转折区后，就进入了饱和区。在这一段，T_2 管、T_4 管均导通并饱和，而 T_3 管、D_4 管截止。输出电压 $U_O\leqslant 0.3V$，为低电平。

由于门槛电平 U_{TH} 所对应的是电压传输特性转折区的中心点，所以在对与非门的简化定性分析中，常以 U_{TH} 为准。认为当 $U_I<U_{TH}$ 时，与非门是关闭的(即 T_2 管、T_4 管截止)；若在其他输入端都为高电平，则当 $U_I>U_{TH}$ 时，与非门导通。与非门关闭时输出高电平，与非门导通时输出低电平。

从电压传输特性曲线上，可以求出如下参数。

(1) 输出高电平 U_{OH}、输出低电平 U_{OL}。

U_{OH} 为与非门关闭时的输出电压，74 系列门电路 U_{OH} 的理论值为 3.6V，产品规定的最小值 $U_{OHMIN}=2.4V$。

U_{OL} 为与非门导通时的输出电压，74 系列门电路 U_{OH} 的理论值为 0.3V，产品规定的最大值 $U_{OLMAX}=0.4V$。

由上述规定可以看出，TTL 门电路输出的高低电平不是一个值，而是一个范围。

(2) 开门电平 U_{ON}、关门电平 U_{OFF}。

在保证输出为额定低电平的条件下所允许的输入高电平的最小值称为开门电平 U_{ON}，常形象地表示为 U_{IHMIN}。

在保证输出为额定高电平的条件下所允许的输入低电平的最大值称为关门电平 U_{OFF}，常形象地表示为 U_{ILMAX}。

为了帮助读者进一步理解 U_{OHMIN}、U_{OLMAX}、U_{IHMIN}、U_{ILMAX} 等参数的含义，依照 TTL 手册"V_{OH} 输出高电平电压——输入端在施加规定的电平下，使输出端为高电平的电压"电特性测试方法的规定，可用实验的方法测量这 4 个参数，各参数间的对应关系如图 3.2.9 所示。由图不难看出，当输入为 U_{IHMIN} 时，对应输出为 U_{OLMAX}；当输入为 U_{ILMAX} 时，对应输出为 U_{OHMIN}。

根据图示关系及相关应用约束，有

$$U_{OHMIN}>U_{IHMIN},\quad U_{OLMAX}<U_{ILMAX}$$

5. 输入端噪声容限

与非门在输入高电平时，为了保证稳定在导通状态，输入的高电平加上瞬态的干扰信号不应小于开门电平 U_{ON}。所以，在输入高电平时，允许的干扰容限为

$$U_{NH}=U_{OHMIN}-U_{IHMIN}$$

式中，U_{NH} 为高电平噪声容限。

与非门输入低电平时，为了保证稳定在关闭状态，输入的低电平加上瞬态的干扰信号不应超过关门电平 U_{OFF}。所以，当输入低电平时，允许的干扰容限为

$$U_{NL} = U_{ILMAX} - U_{OLMAX}$$

式中，U_{NL}为低电平噪声容限。

在集成电路中，用噪声容限的数值来定量地说明门电路的抗干扰能力。显然，噪声容限越大，说明门的抗干扰能力越强。

【例 3.2.1】 如图 3.2.10(a)所示电路中，计算 R_F的最大值。其中的与非门电路 Y_H＝3.4V，Y_L＝0.2V，A_{IHMIN}＝2.0V，A_{ILMAX}＝0.8V，U_{CC}＝5.0V，输入、输出特性分别如图 3.2.4、图 3.2.6、图 3.2.8 所示。

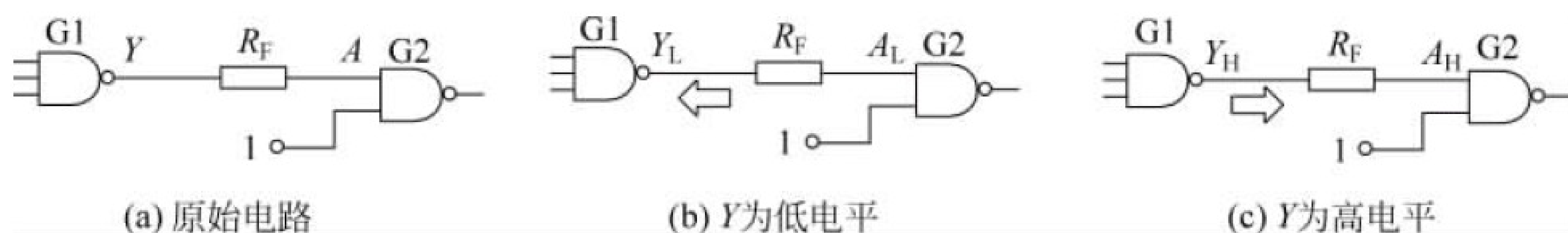


图 3.2.10 例 3.2.1 的图

解 (1) 计算 G1 输出为低电平(Y_L＝0.2V)时 R_F的最大值。

与非门电路输入参考等效电路如图 2.2.3 所示，当 Y 为低电平时，输入电流经电阻 R_F流入 G1，如图 3.2.10(b)所示。

显然，电阻 R_F的接入将使 G2 的输入低电平电位上升，但应满足 $A < A_{ILMAX} = 0.8\text{V}$ 的应用要求，临界输入电流为

$$I_{IL} = \frac{U_{CC} - U_{BEA} - A_{ILMAX}}{R_1} = \frac{5 - 0.7 - 0.8}{4} = 0.875(\text{mA})$$

$$R_{FMAX} = \frac{A_{ILMAX} - Y_L}{I_{IL}} = \frac{0.8 - 0.2}{0.875} \approx 0.686(\text{k}\Omega)$$

(2) 计算 G1 输出为高电平(Y_H＝3.4V)时 R_F的最大值。

当 Y 为高电平时，截止穿透电流经电阻 R_F流入 G2，如图 3.2.10(c)所示。查如图 3.2.4 所示输入特性曲线，U_{IH}＝3.4V 时，I_{IH}＝40μA。

显然，电阻 R_F的接入将使 G2 的输入高电平电位下降，但应满足 $A > A_{IHMAX} = 2.0\text{V}$ 的应用要求，有

$$R_{FMAX} = \frac{Y_H - A_{IHMAX}}{I_{IH}} = \frac{3.4 - 2.0}{40 \times 10^{-6}} = 35(\text{k}\Omega)$$

可见，该电路中，R_F的最大值为 0.686kΩ。

6. 扇出系数 N_O

扇出系数 N_O是指与非门带负载的能力，一般用一个与非门能够带同类门的最大数目表示。对 TTL 与非门有 $N_O \geq 8$，可通过下面的例题来进一步理解。

【例 3.2.2】 在如图 3.2.11 所示电路中，计算与非门 G1 最多可驱动多少个同样的与非门电路。其中的与非门电路输入、输出特性分别如图 3.2.4、图 3.2.6、图 3.2.8 所示，要

求与非门 G1 输出的高、低电平满足 $U_{OH} \geqslant 3.2V$、$U_{OL} \leqslant 0.2V$。

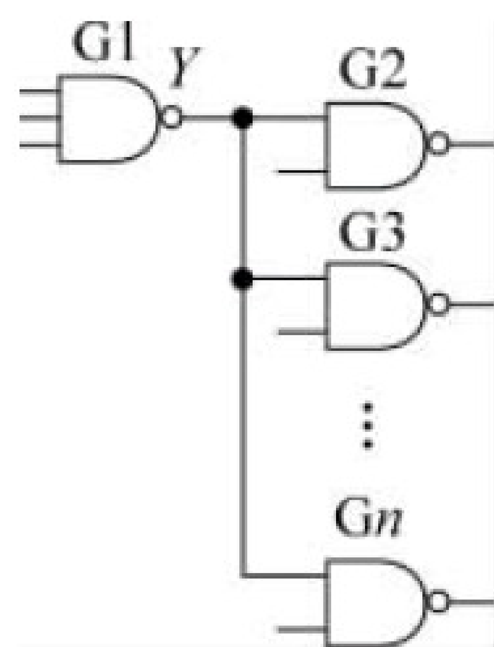


图 3.2.11　例 3.2.2 的图

解　(1) 计算 $U_{OH} > 3.2V$ 时可驱动的与非门数目 N_1。

查如图 3.2.6 所示输出特性曲线，$U_{OH} = 3.2V$ 时，$I_{OH} = -7.5mA$，考虑功耗

$$|I_{OH}| < 0.4mA$$

查如图 3.2.4 所示输入特性曲线，$U_{IH} = 3.2V$ 时，$I_{IH} < 40\mu A$，有

$$N_1 \leqslant \frac{|I_{OH}|}{I_{IH}} = \frac{0.4}{0.04} = 10$$

(2) 计算 $U_{OL} \leqslant 0.2V$ 时可驱动的与非门数目 N_2。

查如图 3.2.8 所示输出特性曲线，$U_{OL} = 0.2V$ 时，$I_{OL} = 16mA$。

查如图 3.2.4 所示输入特性曲线，$U_{IL} = 0.2V$ 时，$I_{IL} = -1mA$，有

$$N_2 \leqslant \frac{I_{OL}}{|I_{IL}|} = \frac{16}{1} = 16$$

可见，该电路中，与非门的扇出系数 $N_O = 10$。

7. 传输延迟时间

在 TTL 电路中，由于二极管和三极管从导通变为截止或者从截止变为导通都需要一定的时间，所以当将理想的矩形电压信号加到 TTL 与非门的输入端时(其他输入端接高电平)，输出电压的波形不仅要比输入信号滞后，而且波形的上升沿和下降沿都变坏，如图 3.2.12 所示。

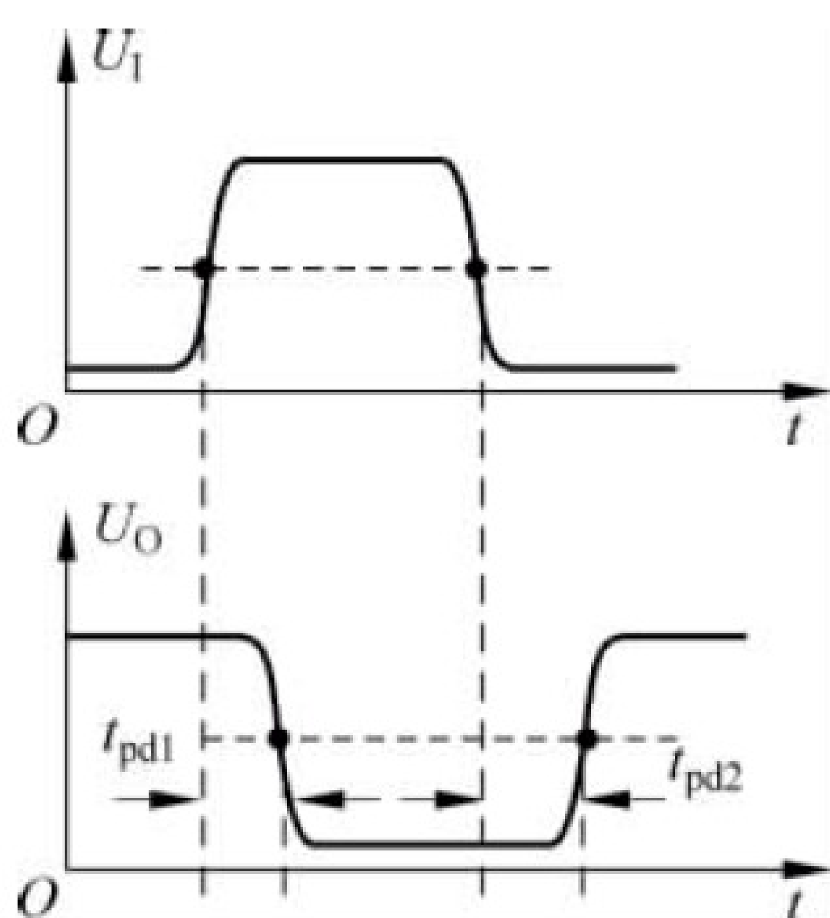


图3.2.12　平均传输延迟时间

通常将输入脉冲上升沿的 50% 处起到输出脉冲下降沿的 50% 处的时间称为上升延迟时间 t_{pd1}；从输入脉冲下降沿的 50% 处起到输出脉冲上升沿的 50% 处的时间称为下降延迟时间 t_{pd2}。t_{pd1} 与 t_{pd2} 的平均值称为平均传输延迟时间，记为 t_{pd}，即

$$t_{pd} = (t_{pd1} + t_{pd2})/2$$

平均传输延迟时间 t_{pd} 是一个很重要的参数，其数值越小说明电路的速度越高。t_{pd} 一般通过实验方法测出，具体值可以从产品手册上查到。

对 74 系列 TTL 门电路而言，输出级的 T_4 管导通时一般工作在深度饱和状态，显然，从深度饱和到截止的时间要长于从截止到饱和的时间，因此，t_{pd2} 略大于 t_{pd1}。

3.2.3　其他逻辑功能的 TTL 门电路

在 TTL 系列集成门电路中，除了与非门外，还有反相器、与门、或门、或非门、异或门、与或非门等几种常见类型的逻辑门电路。必须指出的是，尽管这些门电路逻辑功能不同，

但其输入、输出结构形式基本相同，具有相同的输入、输出特性。

1. 反相器

如图 3.2.13 所示电路为 TTL 反相器的典型电路，是最简单的一种 TTL 门电路。

和与非门相比，TTL 反相器的输入级是一个单发射极的三极管，其余部分和与非门电路相同，逻辑功能显而易见。

2. 或非门

如图 3.2.14 所示电路为 TTL 或非门的典型电路，逻辑功能简要分析如下。

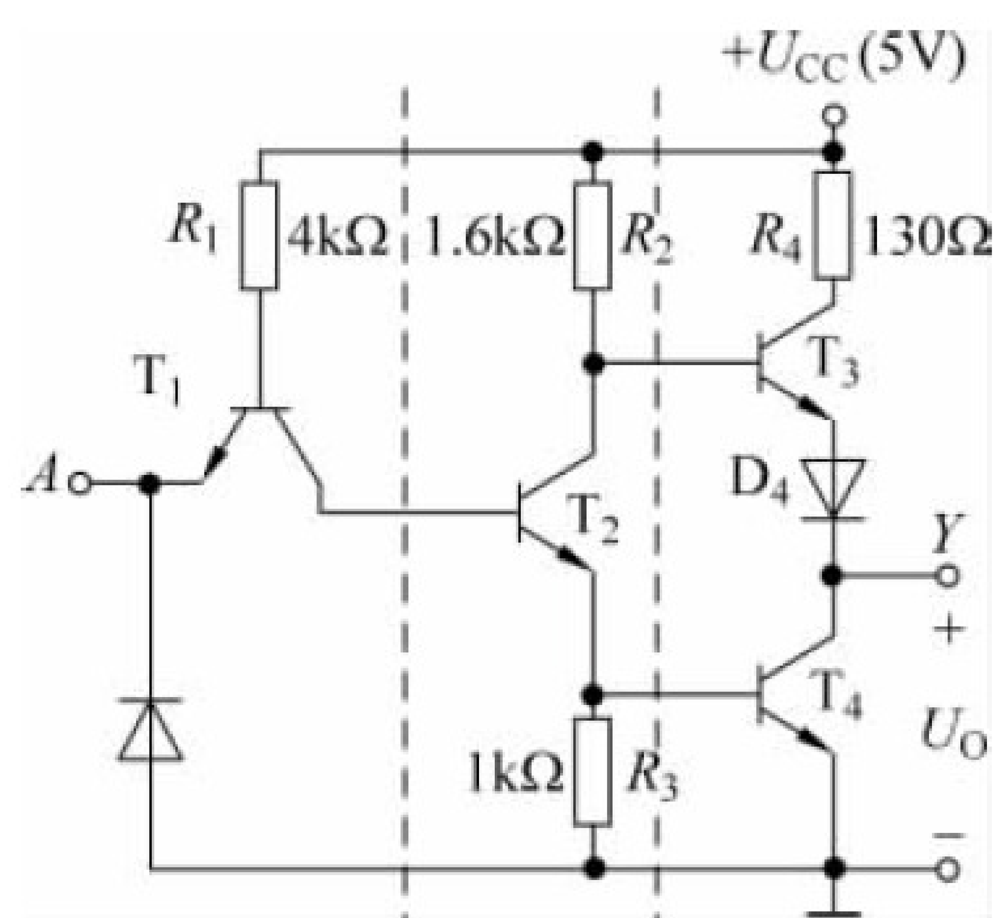


图 3.2.13 TTL 反相器电路

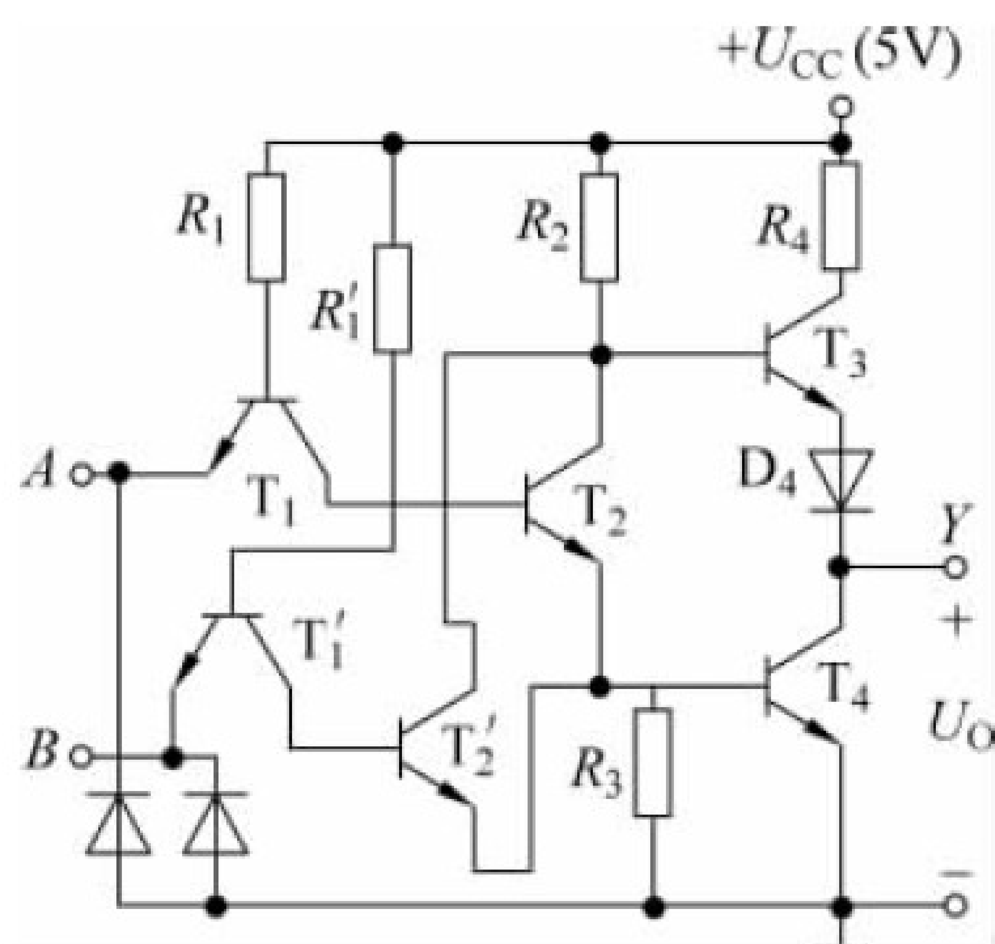


图 3.2.14 TTL 或非门电路

如不考虑输入 B，则输入 A 及相关电路为典型 TTL 反相器电路。类似地，如不考虑输入 A，则输入 B 及相关电路也为典型 TTL 反相器电路。

当 $A=1$ 时，T_4 管饱和导通。此时，如果 B 也为 1，则维持 T_4 管的饱和导通不变；如果 B 为 0，则 T_4 管应截止，T_2' 管也截止。T_2' 管截止，输入 B 对 T_4 管相当于断开，输入 $A=1$ 维持 T_4 管的饱和导通。

根据上面的分析，当 A、B 两个输入中有一个及以上为 1 时，T_4 管饱和导通，输出为 0，实现了或非的逻辑功能。

3. 与或非门

如图 3.2.15 所示电路为 TTL 与或非门的典型电路，逻辑功能简要分析如下。

如不考虑输入 C、D，则输入 A、B 及相关电路为典型 TTL 与非门电路。类似地，如不考虑输入 A、B，则输入 C、D 及相关电路也为典型 TTL 与非门电路。

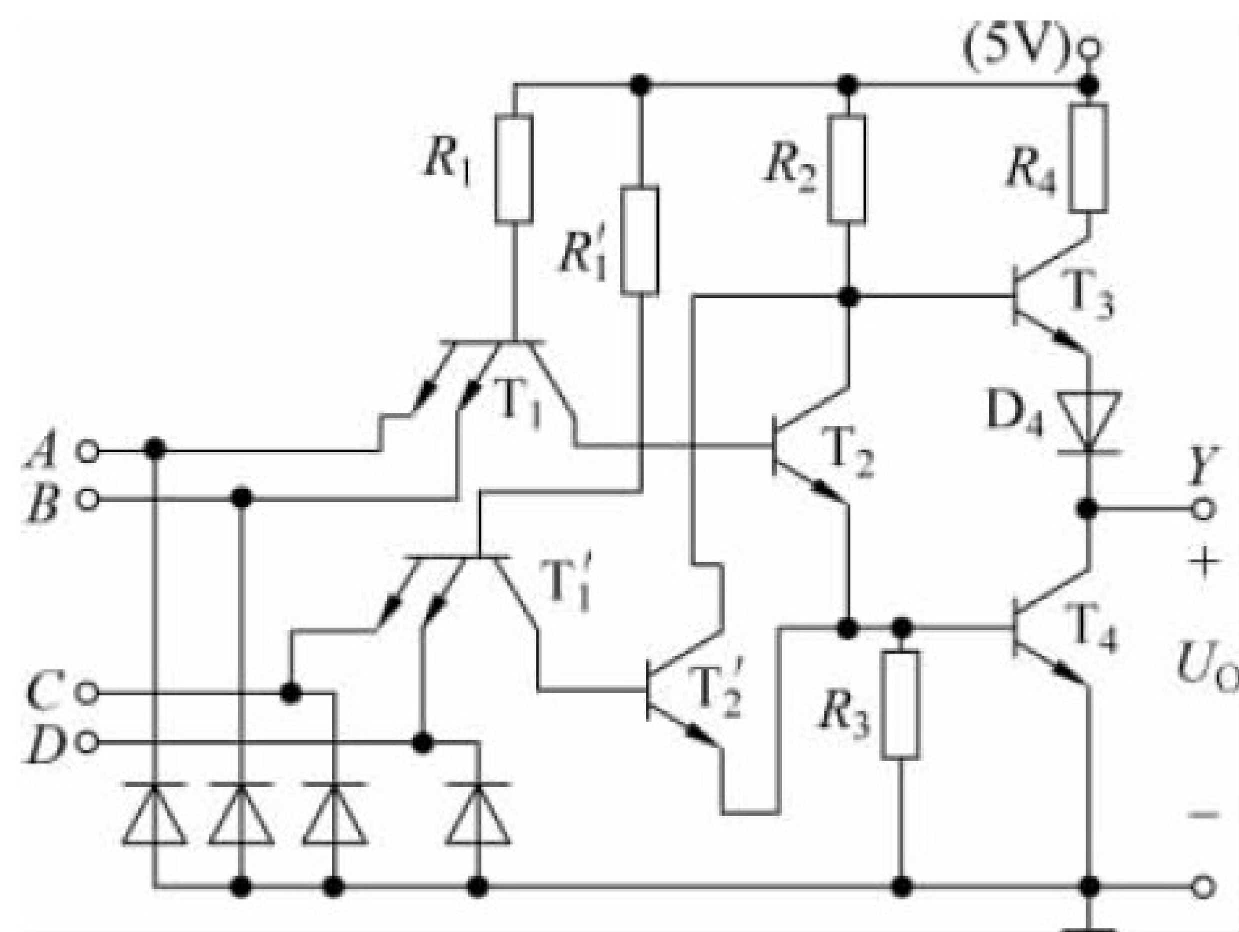


图 3.2.15 与或非门

当 $A=B=1$ 时，T_4 管饱和导通。此时，如果

$C=D=1$,则维持 T_4管的饱和导通不变；如果 C、D 中有 1 个为 0，则 T_4管截止，T_2'管也截止。T_2'管截止，输入 B 对 T_4管相当于断开，输入 $A=B=1$，维持 T_4管的饱和导通。

根据上面的分析，当 AB、CD 两组输入有一组及以上为 1 时，T_4管饱和导通，输出为 0，实现了与或非的逻辑功能，有

$$Y=\overline{AB+CD}$$

关于 TTL 系列其他逻辑功能门电路的典型电路，限于篇幅，请读者参看其他书籍。

3.2.4 其他输出结构的 TTL 与非门



1. 集电极开路输出的 TTL 与非门(OC 门)

如图 3.2.16 所示为集电极开路输出的与非门，如图 3.2.17 所示是它的逻辑符号。必须注意的是，集电极开路输出的与非门必须外接负载电阻 R_C 和电源 U_{CC}'才能正常工作，如图 3.2.16 中虚线部分所示。

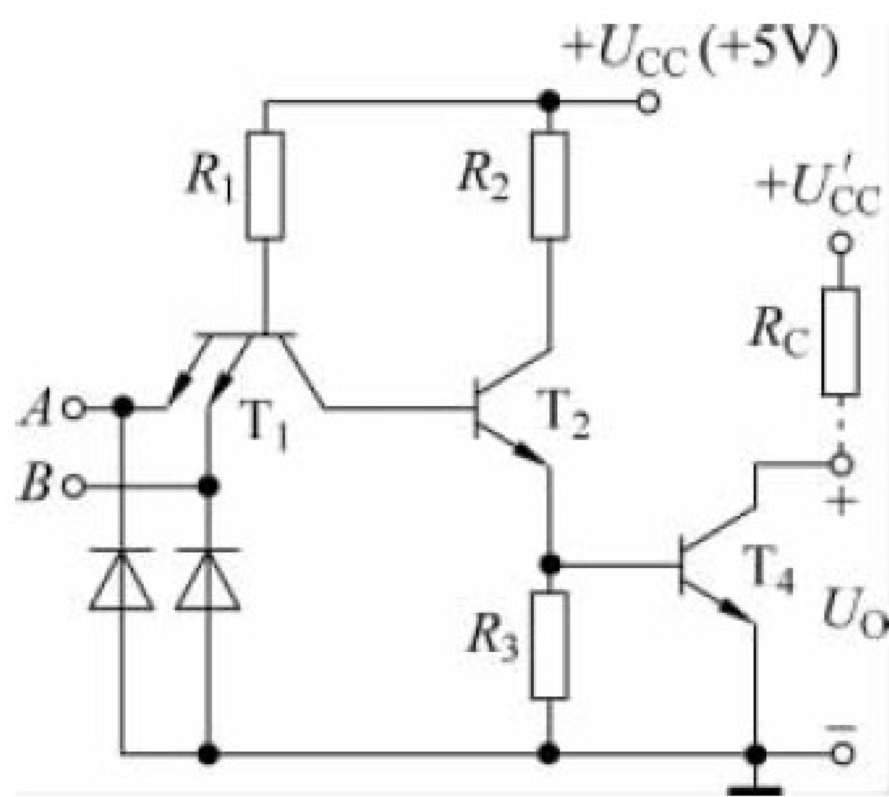


图 3.2.16 OC 与非门

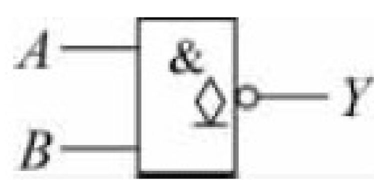


图 3.2.17 OC 与非门逻辑符号

集电极开路的门电路简称为 OC 门。将典型 TTL 与非门电路中的 T_3管、D_4管去掉。如图 3.2.16 所示的 OC 门，当外接 R_C 和电源 U_{CC}'以后，其逻辑功能为 $Y=\overline{AB}$，工作原理十分简单，无须赘述。

由于 OC 门采用外接负载电阻和电源，故可通过选择较高电压的工作电源给负载提供较大的电流，可很好地改善 TTL 门电路的负载能力。可通过下面的例题来进一步理解。

【例 3.2.3】 在如图 3.2.18(a)所示电路中，已知 OC 与非门 G1、G2 输出管截止时漏电流 $I_{OH}=0.2\text{mA}$，导通时允许的最大负载电流 $I_{OLMAX}=16\text{mA}$；与非门 G3、G4、G5 的低电平输入电流 $I_{IL}=-1\text{mA}$，高电平输入电流 $I_{IH}=0.04\text{mA}$，要求 OC 与非门输出的高、低电平满足 $Y_{OH}\geqslant 3.2\text{V}$、$Y_{OL}\leqslant 0.4\text{V}$，计算 R_L的合适值。

解 (1) 计算 OC 与非门 G1、G2 输出为高电平时 R_L的合适值。

输出 $Y=1$，流过 R_L 的电流为 G1、G2 管截止漏电流和 G3、G4、G5 高电平输入电流的和，各电流方向如图 3.2.18(b)所示。

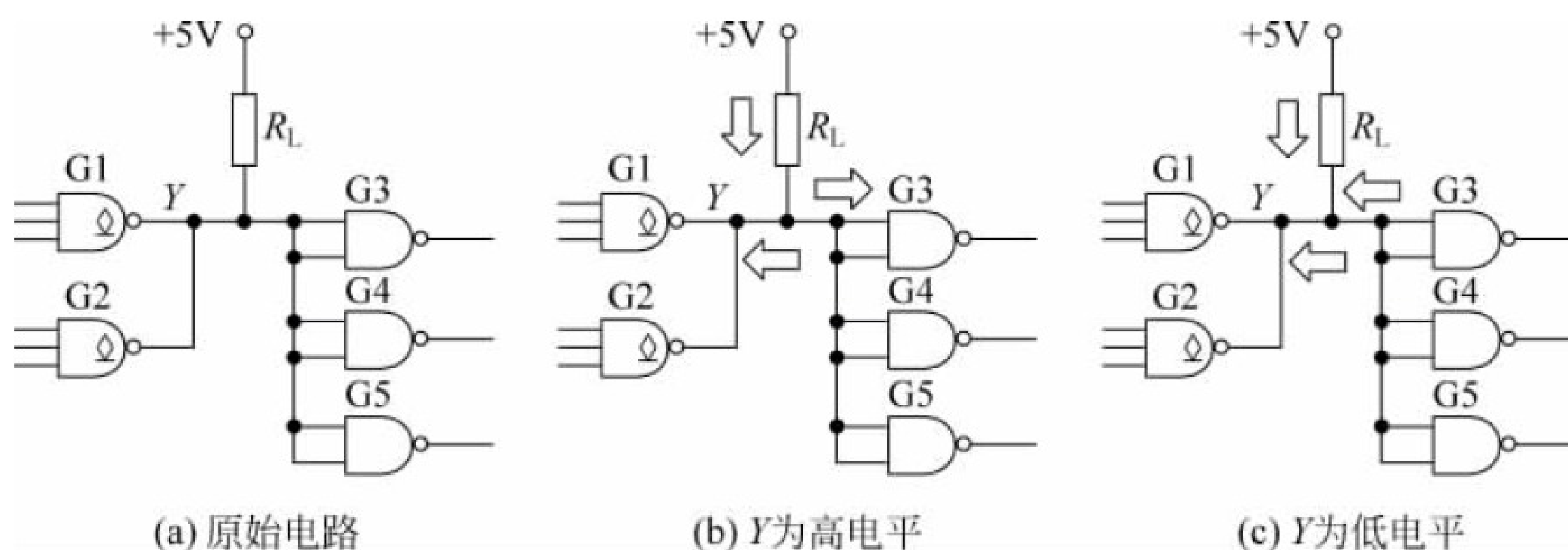


图 3.2.18 例 3.2.3 的图

显然，电阻 R_F 的接入将使 G1、G2 管的输出高电平电位下降，但应满足 $Y_{OH} \geqslant 3.2V$ 的应用要求，流过 R_L 的电流为

$$I_{R_L} = nI_{OH} + mI_{IH} = 2 \times 0.2 + 6 \times 0.04 = 0.64(\text{mA})$$

R_L 的最大值为

$$R_{LMAX} = \frac{5 - Y_{OH}}{I_{R_L}} = \frac{5 - 3.2}{0.64} \approx 2.8(\text{k}\Omega)$$

(2) 计算 OC 与非门 G1、G2 输出为低电平时 R_L 的合适值。

当 Y 为低电平时，各电流示意图如图 3.2.18(c)所示。

显然，电阻 R_F 的减小将使 G1、G2 的输出低电平电位上升，但应满足 $Y_{OL} \leqslant 0.4V$ 的应用要求，流过 R_L 的电流为

$$I_{R_L} = I_{OLMAX} - m\left|I_{IL}\right| = 16 - 3 \times 1 = 13(\text{mA})$$

注意：尽管有两个 OC 与非门，但求其最小值的时候，只能考虑只有一个 OC 门输出为 0 的情况。

此外，由于 G3、G4、G5 的输入端各自短接，参考如图 3.2.2 所示的输入级电路，输入端短接时，相当于一个输入。

R_L 的最小值为

$$R_{LMIN} = \frac{5 - Y_{OL}}{I_{R_L}} = \frac{5 - 0.4}{13} \approx 0.35(\text{k}\Omega)$$

可见，该电路中，R_L 的合适值为 0.35～2.8kΩ，可取

$$R_L = 1\text{k}\Omega$$

必须指出的是，应用实践中，输入可以短接，输出一般是不可以短接的。在如图 3.2.18 所示电路中，将与非门 G1、G2 的输出直接短接构成一个输出 Y，之所以可以将输出直接短接，原因是 G1、G2 门为 OC 门。

两个 OC 门输出直接短接参考电路如图 3.2.19 所示。图中，当 $A=B=1$ 时，OC 门 G1 的 T_4 管饱和导通，输出 $Y_0=0$。

此时，如果 $C=D=1$，OC 门 G2 的 T_4 管也饱和导通，输出 Y_1 也为 0，总体输出 $Y=0$；如果 C、D 至少有一个为 0，则 OC 门 G2 的 T_4 管截止，输出 $Y_1=1$，但由于此时 $Y_0=0$，而 G2

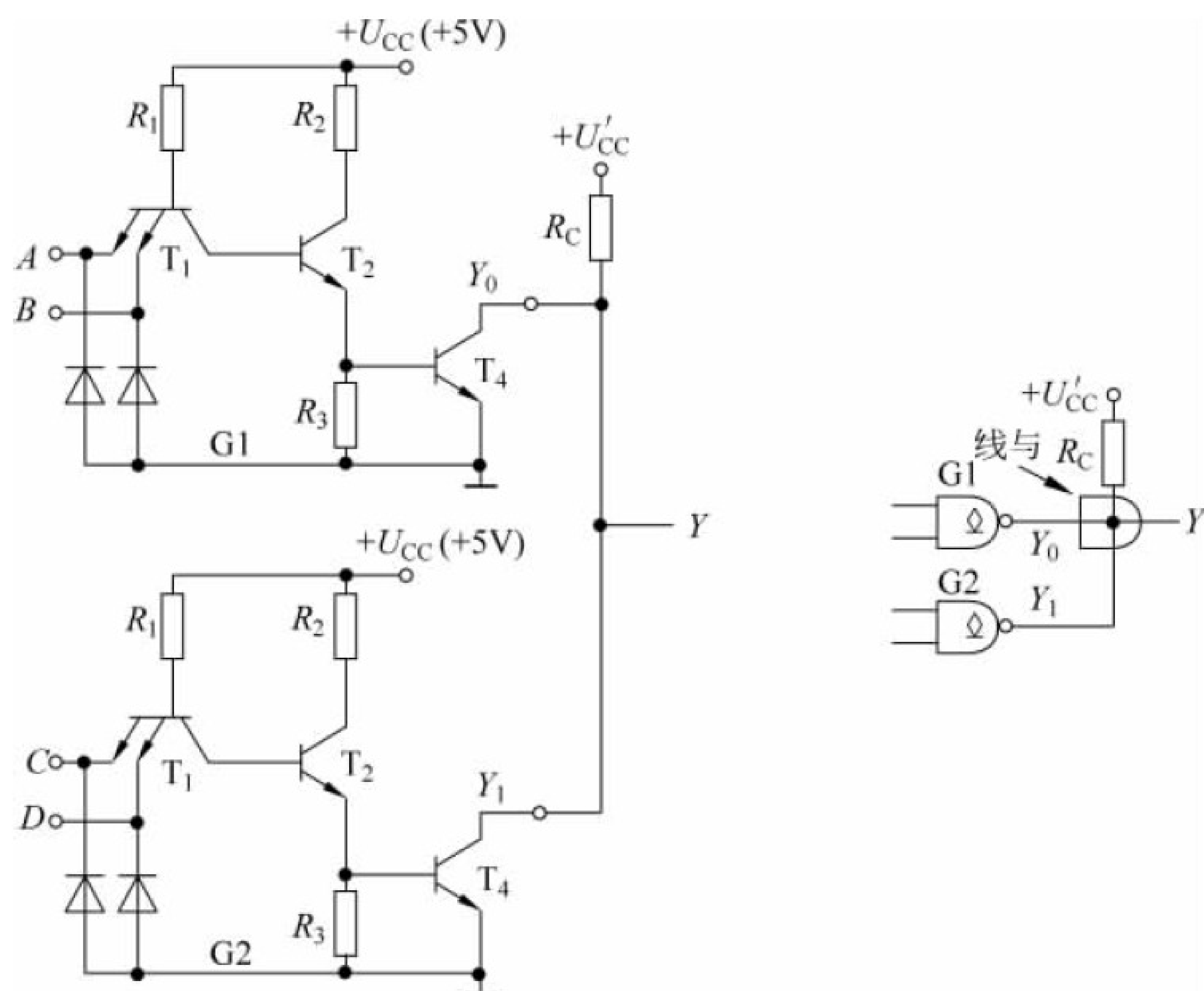


图 3.2.19　OC 门输出直接短接示意图

的 T_4 管截止后，G2 的输出 Y_1 可理解为与 G2 的 T_4 管断开，总体输出 Y 依然为 0。

可见，两个 OC 门输出直接短接，相应的输出 Y_0、Y_1 至少有一个为 0，总体输出为 0，具有与的逻辑功能，把这种通过输出线直接短接实现的与逻辑功能形象地称为“线与”。

可见，图 3.2.19 所示电路的输出

$$Y=\overline{AB}\ \overline{CD}=\overline{AB+CD}$$

2. 三态输出与非门(TS 与非门)

输出三态门简称三态门(TS 门)，它是在普通门的基础上增加控制端和控制电路组成的。

如图 3.2.20 所示为三态门的电路原理图，其逻辑符号如图 3.2.21 所示。三态门的逻辑功能为与非的逻辑功能。

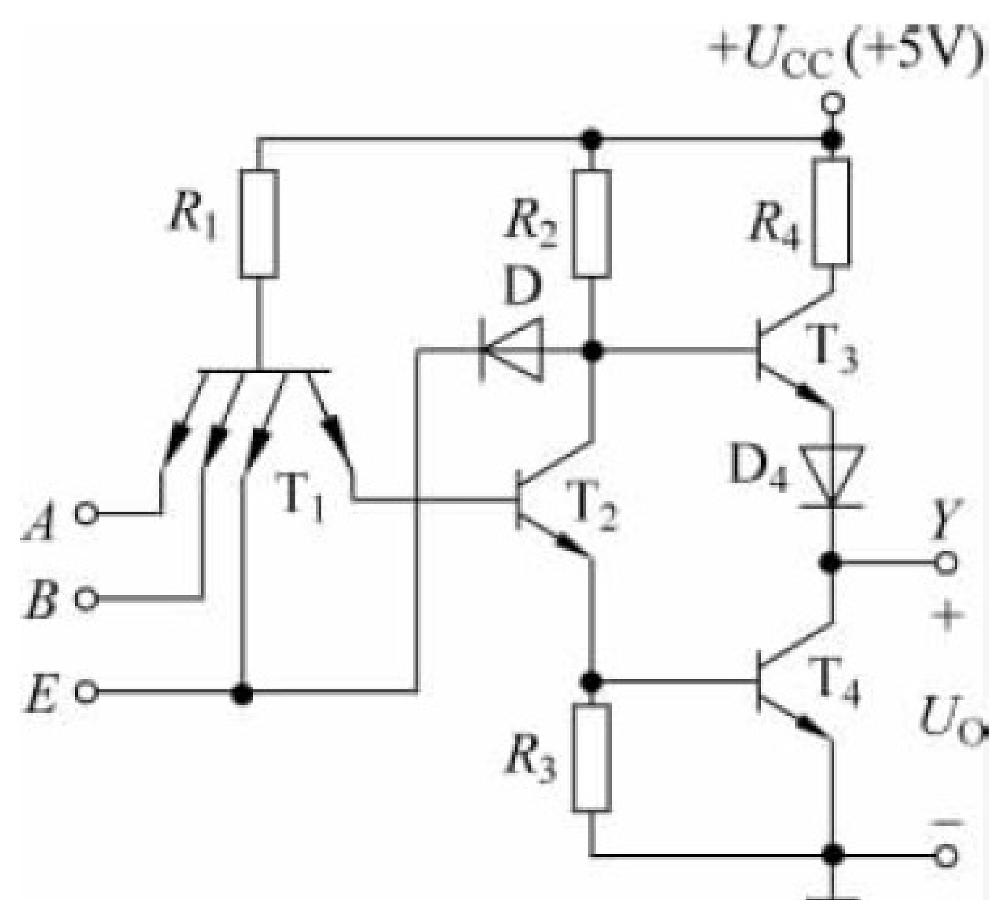


图 3.2.20　三态输出与非门

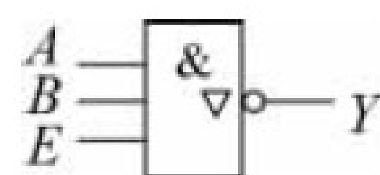


图 3.2.21　TS 门逻辑符号

在如图 3.2.20 所示电路中，当使能控制信号 $E=0$ 时，T_1管导通，T_2管、T_4管均截止，而导通的二极管 D 将 T_2管集电极电位钳制在小于或等于 1V 的电平上，使 T_3管和 D_4管也不能导通。此时，输出端 Y 对电源 U_{CC}和对地都是断开的，呈现为高阻抗状态，记为

$$Y = Z$$

当使能控制端信号 $E=1$ 时，D 管截止。此时，三态门处于工作状态，即有

$$Y = \overline{ABE}$$

综上所述，图 3.2.20 所示电路的输出端有三种状态：高电平、低电平、高阻抗。而处于工作状态时，实现的功能又是与非逻辑运算，所以该电路称为 TTL 输出三态与非门。

在数字系统中，为了减少输出连线，经常在一条数据总线上分时传递若干门电路的输出信号，利用三态门可以实现这种总线结构，如图 3.2.22 所示。

在图 3.2.22 中，只要控制各个门的 E 端，则可实现在一条数据总线上分时传递各门电路的输出信号。例如现在要传递门 Gi 的输出信号，则令使能控制端 $E_i=1$，使三态门 Gi 工作；而其他三态门的使能控制端 E 均为 0，输出端为高阻抗状态，不工作；这样就将三态门 Gi 的输出信号送到了总线上。

在 TTL 电路中，不仅有三态输出的与非门、反相器、缓冲器等，而且在许多中规模乃致大规模集成电路中也采用了三态输出电路。

此外，还可利用三态输出缓冲器实现双向输出的总线结构。用三态缓冲器实现的 1 位数据双向输出总线结构如图 3.2.23 所示。图中，三态门 G1 的输出控制端为高电平有效，G2 的输出控制端为低电平有效。当 $E=1$ 时，G1 工作，G2 不工作，数据 D_0被送往数据总线；当 $E=0$ 时，G2 工作，G1 不工作，从数据总线上读取数据并送给 D_1。

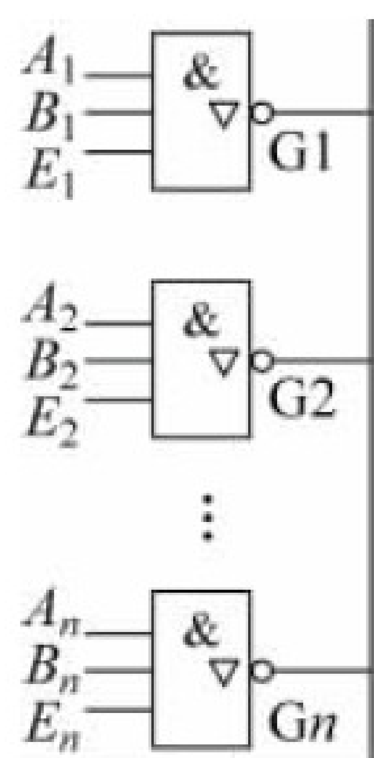


图 3.2.22　将三态与非门接成总线结构

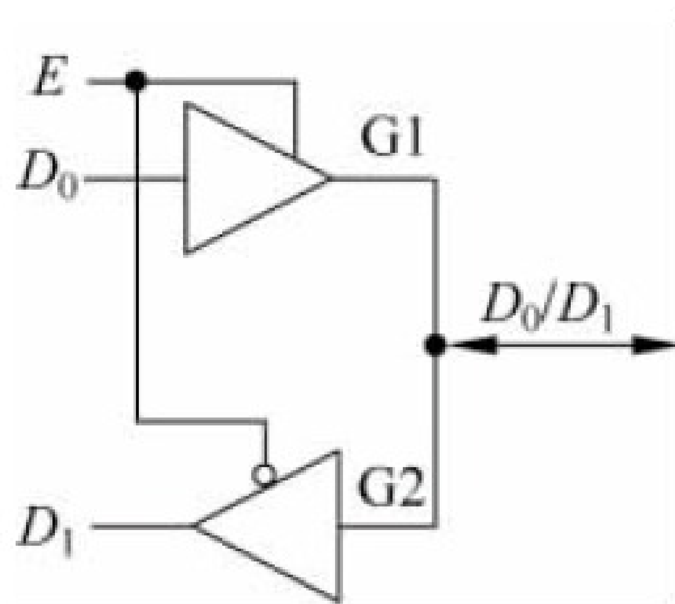


图 3.2.23　用三态输出缓冲器实现双向输出

3.2.5　TTL 数字集成电路的各种系列

TI 公司最初生产的 TTL 电路取名为 SN54/74 系列，为 TTL 基本系列。54 系列和 74 系列产品的主要区别是允许的环境工作温度不同。54 系列产品允许的环境工作温度为 $-55\sim+125$℃，而 74 系列产品允许的环境工作温度为 $-40\sim+85$℃。在后面的内容中，统一使用 74 系列，读者应注意二者之间的差异。

74 系列之后，又相继生产了 74H、74L、74S、74LS、74AS、74ALS、74F 等改进系列。

74 系列电路特点如前所述，每级门电路的传输时间约为 9ns，功耗约为 10mW，速度、功耗等方面的性能均不能令人满意。74H(High-speed TTL)、74L(Low-power TTL)系列是早期采用的两种改进系列。

74H 系列将输出电路中的二极管改为三极管，采用了两级射极输出；并将所有的电阻值减小许多，从而提高了负载能力，同时也提高了工作速度。其平均传输延迟时间 t_{pd} 约为 74 系列与非门的平均传输延迟时间的 1/2。因为 74H 系列电路中电阻值大大减小，将使内部电流增加，故增加了电路的功耗。74L 系列加大了各电阻的阻值，降低了电路的功耗，但增加了电路的传输延迟时间。这两个系列产品的综合性能并未明显改善，已经被淘汰。

74 系列电路和 74H 系列电路中的三极管在导通时几乎都处于饱和状态，当晶体三极管由饱和状态转换到截止状态时，需要一个较大的存储时间以消除在晶体管基区内新存储的电荷，所以这是限制速度不能提高的主要原因。74S(Schottly TTL)系列又称为肖特基系列，电路采用了抗饱和电路，在三极管的基极和集电极间串接一只肖特基二极管(抗饱和三极管)，如图 3.2.24 所示，有效避免了三极管饱和时进入深度饱和状态，提高了工作速度，平均传输时间约为 3ns。

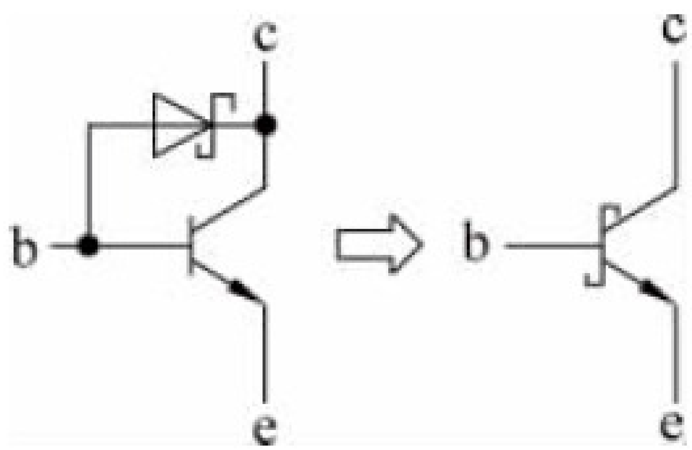


图 3.2.24 抗饱和三极管

74LS 系列电路称为低功耗肖特基系列，大幅增加了电路中各电阻的阻值，降低了功耗。74LS 系列门电路功耗仅为 74S 系列的 1/3、74 系列的 1/5、74H 系列的 1/10。因为 74LS 系列门电路使用了抗饱和三极管，有利于缩短传输延迟时间，因此，尽管大幅增加了电路中各电阻的阻值，传输延迟时间依然可以达到 74 系列的水平，是目前依然在使用的 TTL 集成门电路，也是 TTL 的主流系列。

74AS(Advanced Schottly TTL)系列又称为高级肖特基系列，结构与 74LS 系列相似，但大幅度减小了电路中各电阻的阻值，提高了工作速度，缺点是功耗较大，比 74S 系列的功耗还要大一些。

74ALS 是为了获得更小的延迟-功耗积而设计的改进系列。它的延迟-功耗积是所有 TTL 系列中最小的一种。74F(Flash TTL)系列在功耗和速度两方面均介于 74AS 和 74ALS 系列之间，为设计人员提供了一种折中选择。

TTL 电路不同系列的四二输入与非门(74××00)的主要性能比较如表 3.2.3 所示，读者可参考 TI 公司产品的性能参数理解其他半导体器件公司的类似产品及其之间的差异。

表 3.2.3 TTL 系列器件(74××00)主要性能比较

参数名称与符号	系列					
	74	74S	74LS	74AS	74ALS	74F
U_{ILMAX}/V	0.8	0.8	0.8	0.8	0.8	0.8
U_{OLMAX}/V	0.4	0.5	0.5	0.5	0.5	0.5
U_{IHMIN}/V	2.0	2.0	2.0	2.0	2.0	2.0

续表

参数名称与符号	系　列					
	74	74S	74LS	74AS	74ALS	74F
U_{OHMIN}/V	2.4	2.7	2.7	2.7	2.7	2.7
I_{ILMAX}/mA	−1.0	−2.0	−0.4	−0.5	−0.2	−0.6
I_{OLMAX}/mA	16	20	8	20	8	20
I_{IHMAX}/μA	40	50	20	20	20	20
I_{OHMAX}/mA	−0.4	−1.0	−0.4	−2.0	−0.4	−1.0
t_{pd}/ns	9	3	9.5	1.7	4	3
功耗/mW	10	19	2	8	1.2	4
延迟-功耗积	90	57	19	13.6	4.8	12

*3.2.6 ECL 集成电路简介

发射极耦合逻辑(Emitter Coupled Logic,ECL)电路中的信号传递主要通过每个三极管的集电极电流信号逐级传递进行,也称电流型逻辑(Current-Mode Logic)电路。

ECL 集成电路主要优点如下:

(1) 正常工作状态下,ECL 电路中的晶体管工作于线性区或截止区,为非饱和型逻辑电路。

(2) ECL 电路中的晶体管工作在线性区、截止区两个工作区,其逻辑摆幅较小。由于逻辑摆幅小,导通时未进入饱和状态,因此有效缩短了电路各节点电位的上升时间和下降时间。目前,ECL 门电路的传输延迟时间已缩短至 0.1ns 以内,是所有逻辑门电路中最快的一种,常用于高速系统中。

(3) ECL 集成电路一般采用互补输出,输出开关管对为射极跟随输出,发射极反馈电阻也比较大,故这种电路具有很低的输出阻抗,负载特性好。国产 CE10K 系列门电路的扇出系数达 90 以上。

尽管 ECL 门电路具有负载能力强、速度快的优点,其缺点也非常明显的,主要有:

(1) 三极管导通时工作在线性区,三极管 CE 极压降高,功耗很大,每个门的平均功耗可达 100mW 以上。

(2) 逻辑摆幅小,输出电平的稳定性差,噪声容限低,对抗干扰能力不利。

基于上面的优缺点,虽然 ECL 电路与 TTL 电路几乎同时研发成功并投入使用,但其应用范围和普及程度远不如 TTL 电路,产品的种类也不如 TTL 丰富。关于 ECL 电路的更多知识,请读者参考相关专业书籍。

复习与思考

在如图 3.2.25 所示电路中,图(a)是或非门,图(b)是三态与非门,请参考此电路设计一个三态或非门。

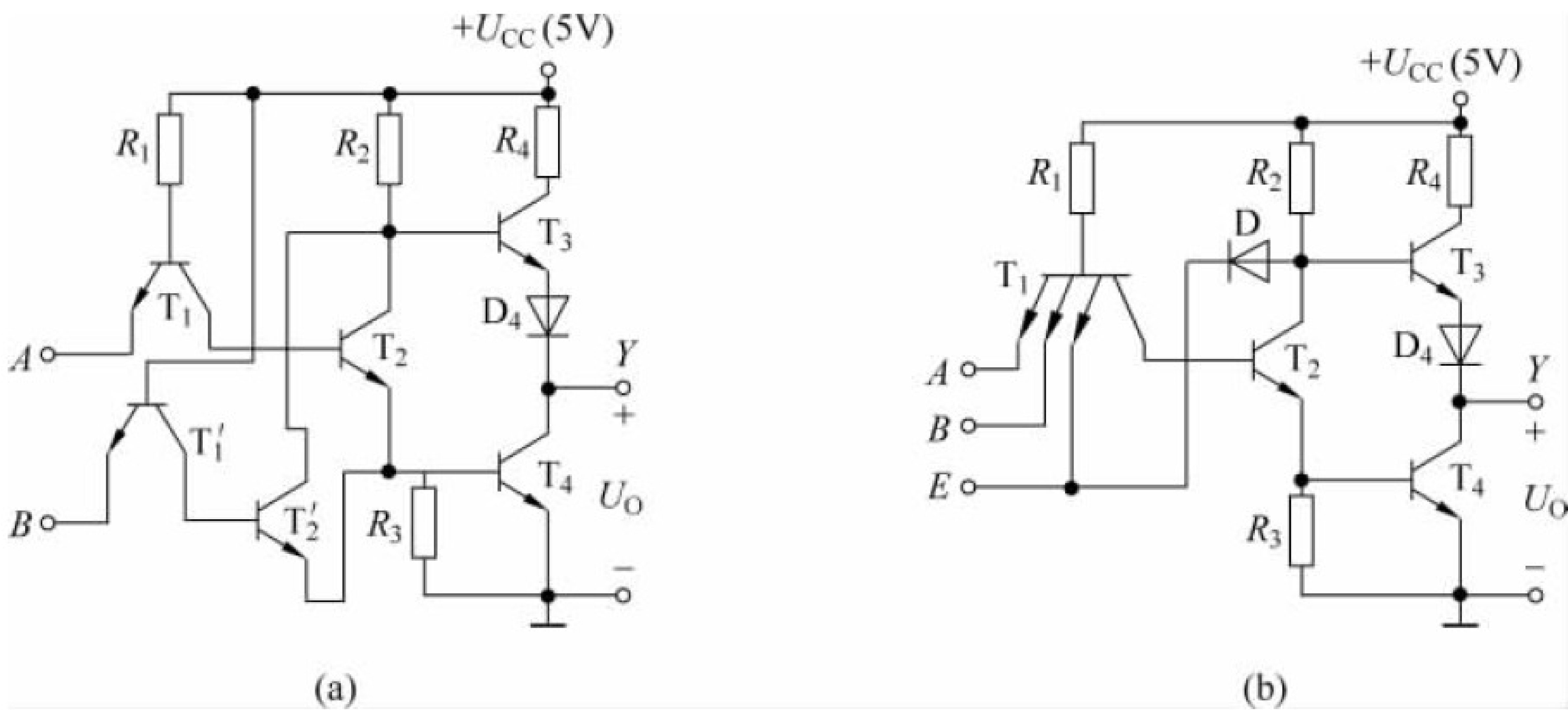


图 3.2.25　复习与思考的图

3.3　利用 MOS 管构成逻辑门电路

MOS 管（Metal-Oxide-Semiconductor Field-Effect Transistor，金属-氧化物-半导体场效应管）用来构成常见逻辑门电路。

场效应管(FET)是利用输入回路的电场效应来控制输出回路电流的一种半导体器件，是一种电压控制器件，理想情况下输入电流为零，即功耗为零。与以三极管为开关器件的 TTL 电路相比，MOS 电路具有输入电阻大、功耗低、噪声低等突出优点，已成为制造超大规模集成电路的主要器件，在很多场合下取代了 TTL 电路。



3.3.1　MOS 管的开关特性

场效应管是一个电压控制器件，结构示意图如图 3.3.1(a)所示。三个与外部连接的电极，分别为源极(S，相当于三极管的发射极)、栅极(G，相当于三极管的基极)、漏极(D，相当于三极管的集电极)。

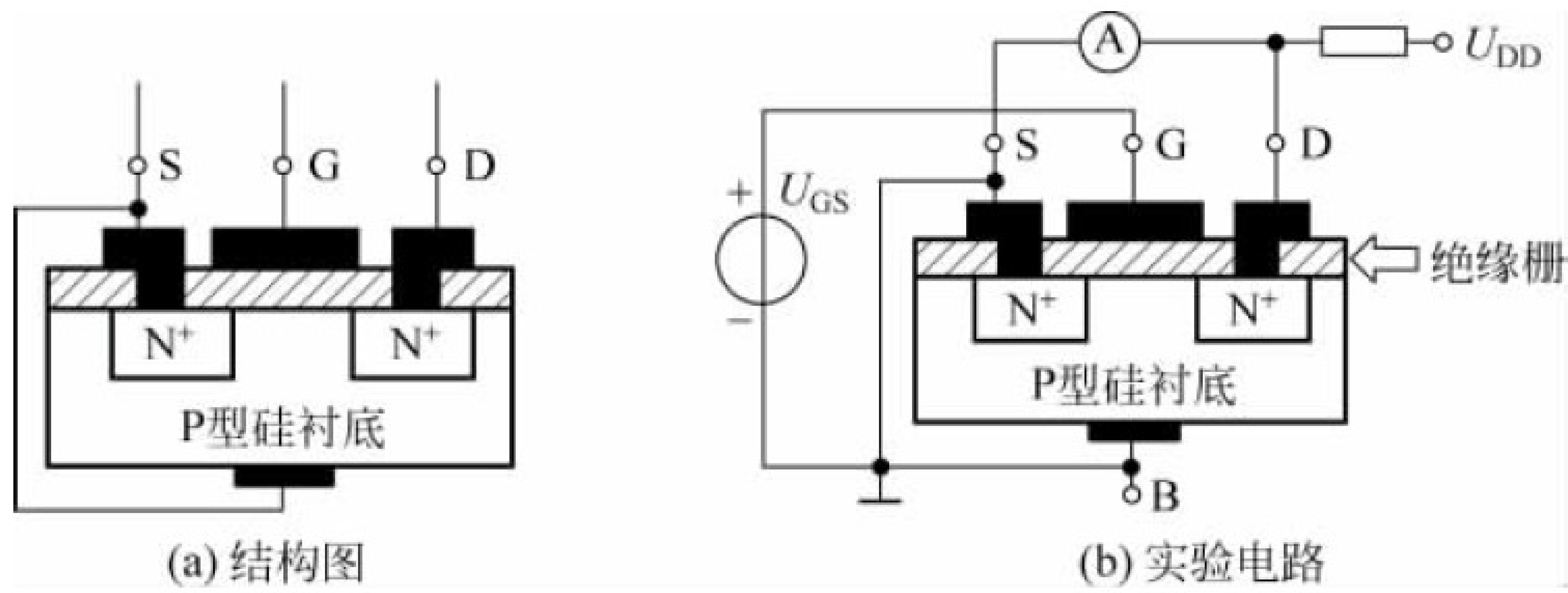


图 3.3.1　MOS 管的图

从制作工艺角度，为满足电压控制特性的需要，一般在一定结构的半导体器件上，加上二氧化硅和金属，形成与外部连接的电极，这样的场效应管称为 MOS 型场效应管。MOS

型 FET 的衬底(B)一般和源极相连,大多数情况下,源极和漏极可以互换。

可通过如图 3.3.1(b)所示的实验电路来理解场效应管的电压控制特性。图中,U_{GS}为可调电源,可从$-U_{DD}$到U_{DD}连续调节。

当U_{GS}为$-U_{DD}$时,电流表指示为 0,基本没有电流流过,继续调大U_{GS},电流表指示依旧为 0,参考效果如图 3.3.2(a)所示。

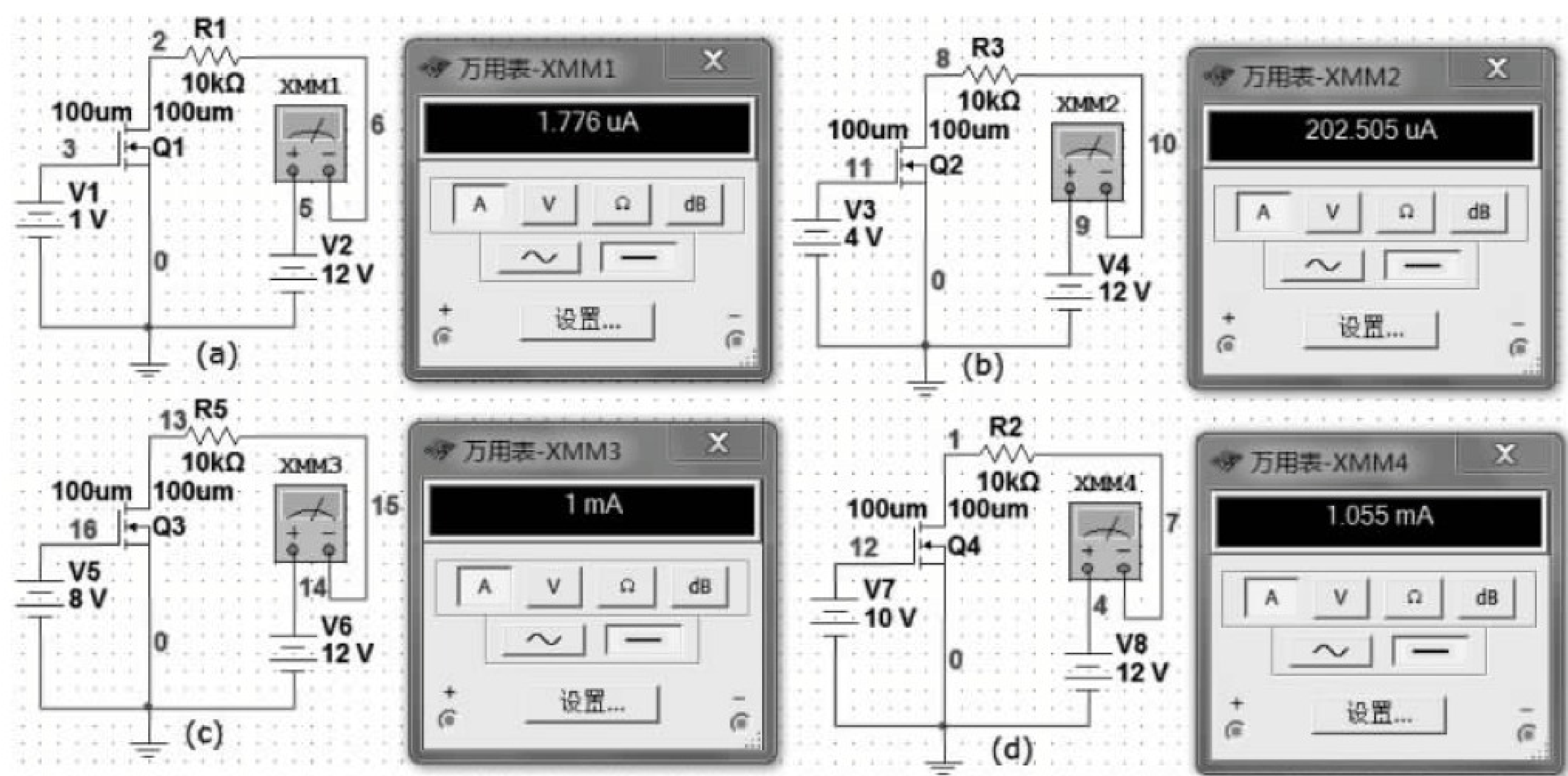


图 3.3.2 场效应管的电压控制特性仿真电路

继续调大U_{GS},当U_{GS}大于某个值时,电流表有明显指示,继续调大U_{GS},电流表上流过的电流相应增大,这便是场效应管电压控制输出回路电流特性,参考效果如图 3.3.2(b)、(c)所示。

仔细分析如图 3.3.1 所示的实验电路,当U_{GS}的值为负或较小时,电流表指示为 0,主要原因是因为 D、S 之间是两个背靠背的二极管。U_{GS}加上正的电压,当达到一定值时,在电动势的作用下,改变内部载流子的分布,在 D、S 之间形成一个导电通路,场效应管导通。

当U_{GS}的值超过一定大小时,U_{GS}增大,电流表上流过的电流基本不变,参考效果如图 3.3.2(c)、(d)所示。

在如图 3.3.1(b)所示的实验电路中,D、S 之间的导电通路为 N 型(称为 N 型沟道),只有当U_{GS}大于某个值时,电流表才有指示,场效应管导通,对应的 MOS 类型为 N 沟道增强型 MOS 管(NEMOS)。

可这样理解 P 型衬底的 MOS 管导电通路为 N 型的原因:由于两个 N 型区为高掺杂区,PN 结在反向电场的作用下,反向漂移加强。当反向电场加强到一定程度时,将在 D、S 之间形成一个 N 型的导电通路。

类似地,可选择 N 型衬底的 MOS 管,两个 P 型区为高掺杂区,D、S 之间的导电通路为 P 型,其导通方式与实验方式则有所区别,读者可结合 PN 结的特点理解 P 沟道增强型 MOS 管(PEMOS)的电压控制特点,具体为:U_{GS}加负的电压,当绝对值达到一定值时,D、S

之间形成一个导电通路，场效应管导通，对应的 MOS 类型为 N 沟道增强型 MOS 管(NEMOS)。

根据制作工艺及应用需求，对 N 沟道场效应管，存在 U_{GS}大于某个负电压时可导通、U_{GS}大于零时可导通、U_{GS}大于某个正电压时可导通等类型，包括结型、耗尽型、增强型 3 种。场效应管的系列树如图 3.3.3 所示，不同类型场效应管的电路符号如图 3.3.4 所示。

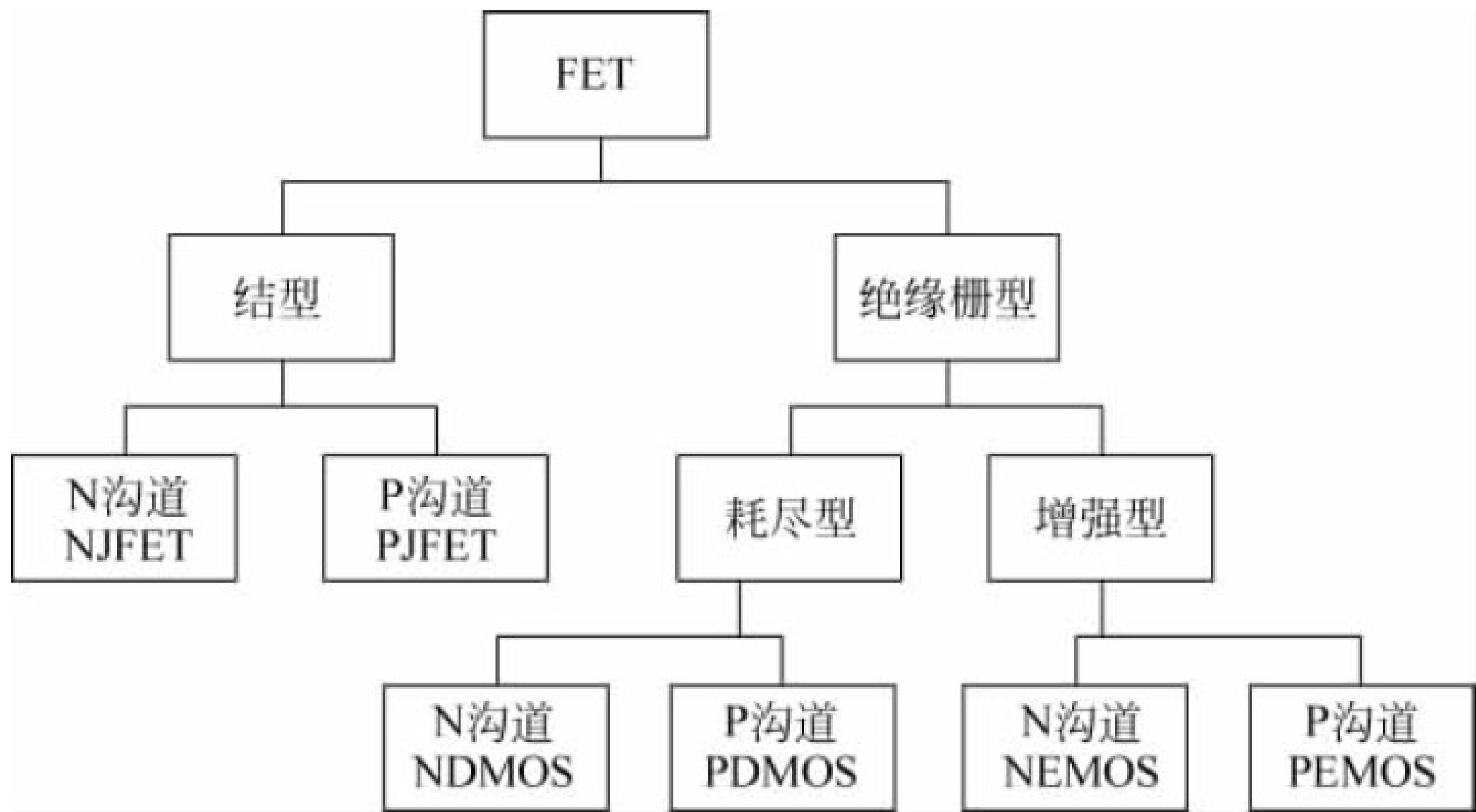


图 3.3.3 场效应管的系列树

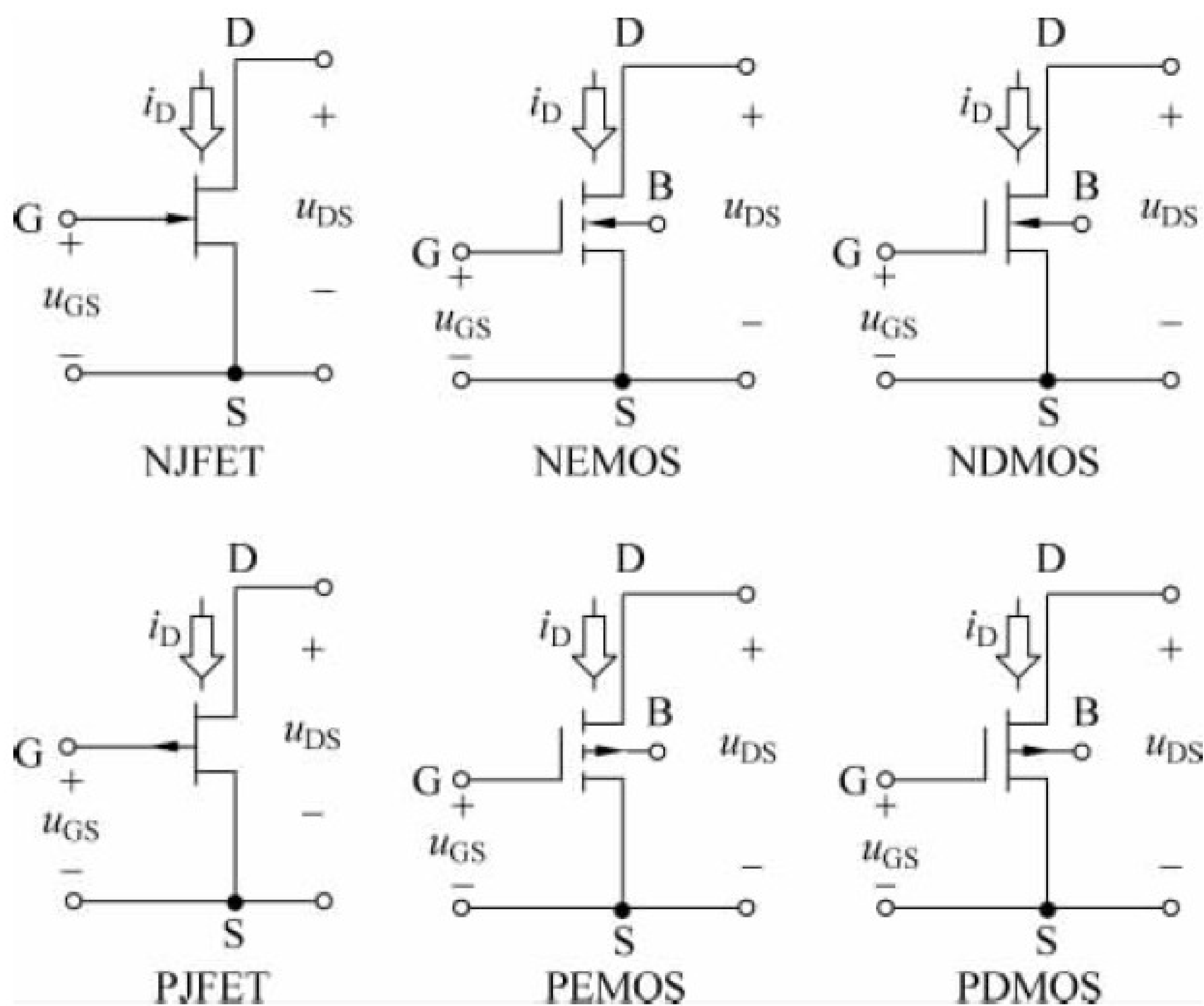


图 3.3.4 场效应管的电路符号

在 CMOS 集成电路中，使用 PEMOS、NEMOS 两种类型的 MOS 管作为开关原件，也常使用 PMOS、NMOS 等术语描述 MOS 管，当使用这些术语描述时，均指增强型的 MOS 管。

可通过如图 3.3.1(b)所示的实验电路进一步理解 MOS 管的输入输出特性。图中，MOS 管导通时，G、S 间经过绝缘栅后相互连接，输入电流 $i_G \approx 0$，输入特性显而易见。

输出特性常用输出电压、输出电流等参数来描述。在如图 3.3.1(b)所示的实验电路中，输出电流与控制电压栅即源电压 U_{GS}相关。N 沟道增强型 MOS 场效应管的输出、转移

特性如图 3.3.5 所示。

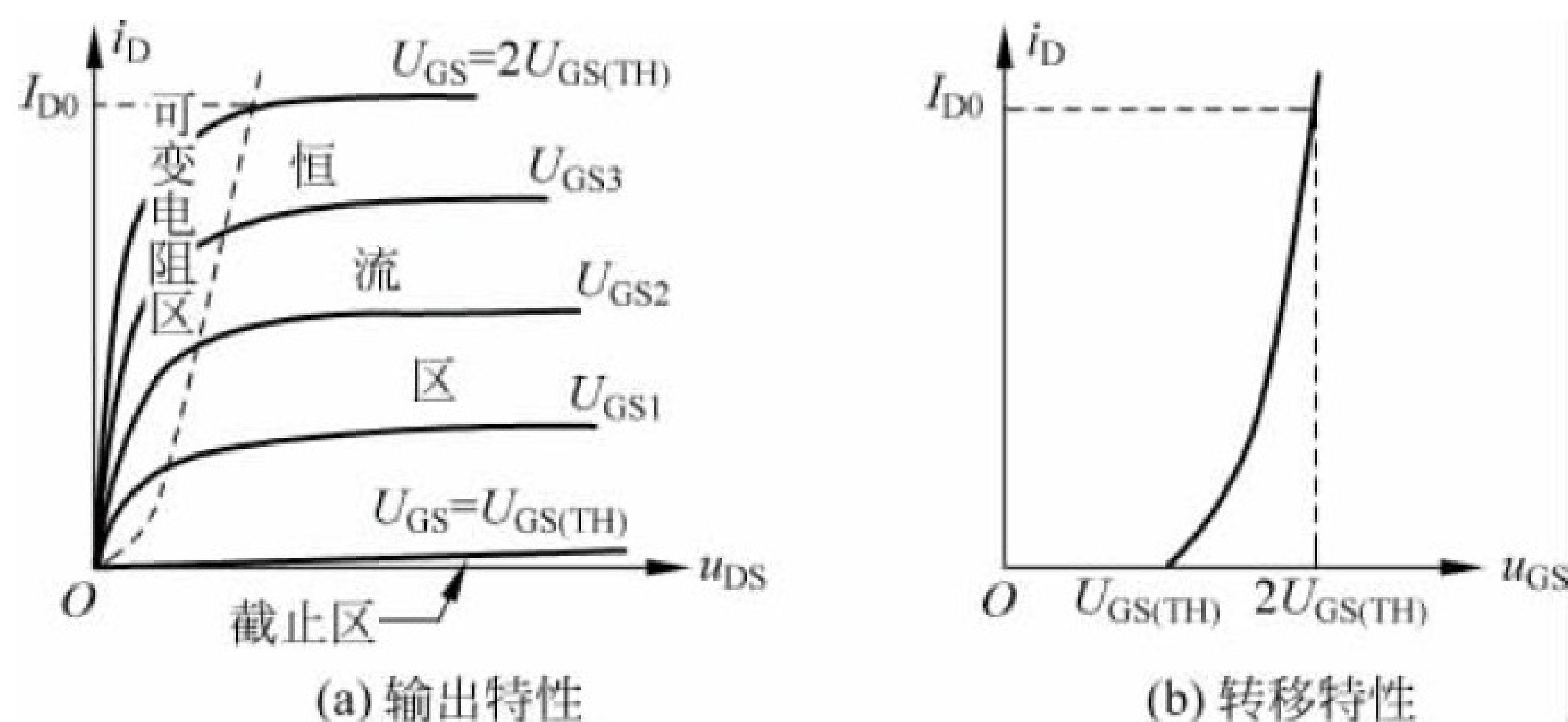


图 3.3.5　N 沟道增强型 MOS 场效应管特性

由如图 3.3.5(a)所示的输出特性曲线，可知 MOS 管也包括 3 个工作区，分别为可变电阻区、恒流区、截止区，类似三极管的饱和、放大、截止 3 个区。

在其中的可变电阻区，电流几乎直线上升，相当于开关闭合。在截止区，几乎没有电流，相当于开关断开。

适当选择电路参数，在数字电平作用下，可使 MOS 管在可变电阻、截止两个区工作，可见，MOS 管可作为开关使用。

图 3.3.1(b)所示的实验电路用 MOS 电路符号表示，其余电压、电流用输入、输出等形式表示，如图 3.3.6 所示。

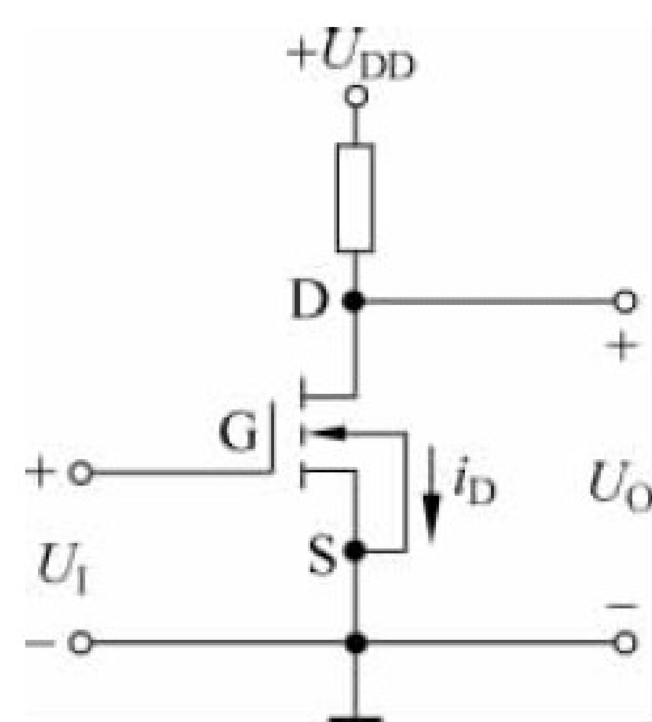


图 3.3.6　MOS 管非门

根据上面的分析，当 $U_I=1$ 时，大于 MOS 管开启电压，场效应管导通，工作在可变电阻区，$U_O=0$，为低电平。当 $U_I=0$ 时，小于 MOS 管开启电压，场效应管截止，工作在截止区，输出电流近似为 0，$U_O=1$，为高电平。可见，图 3.3.1(b)所示的实验电路构成了 MOS 非门电路。

3.3.2　常用 CMOS 门电路构成特点

尽管单个 MOS 管便可构成非门电路，但该电路的输入、输出特性较差。应用实践中的 MOS 门电路主要为 CMOS 门电路。CMOS 门电路是由 P 沟道增强型 MOS 管和 N 沟道增强型 MOS 管按照互补对称形式连接起来构成的逻辑运算电路(并由此而得名)。

CMOS 门电路中，最常用的门电路有反相器和传输门。

1. CMOS 反相器

将一个 P 沟道增强型 MOS 管和一个 N 沟道增强型 MOS 管串联互补，就组成了一个 CMOS 反相器。一般以 P 沟道 MOS 管作为负载管，N 沟道 MOS 管作为输入管，其电路如图 3.3.7(a)所示。图中，它们的栅极 G_N、G_P 连接起来作为反相器的输入端，漏极 D_N、D_P 连接起来作为反相器的输出端，T_P 管的源极 S_P 接电源 U_{DD}，T_N 管的源极 S_N 接地。T_N 管、T_P 管

的特性对称，$U_{GS(TH)N}>0$，$U_{GS(TH)P}<0$，$U_{GS(TH)N}=|U_{GS(TH)P}|$。为了保证电路能够正常工作，要求 $U_{DD}>U_{GS(TH)N}+|U_{GS(TH)P}|$。

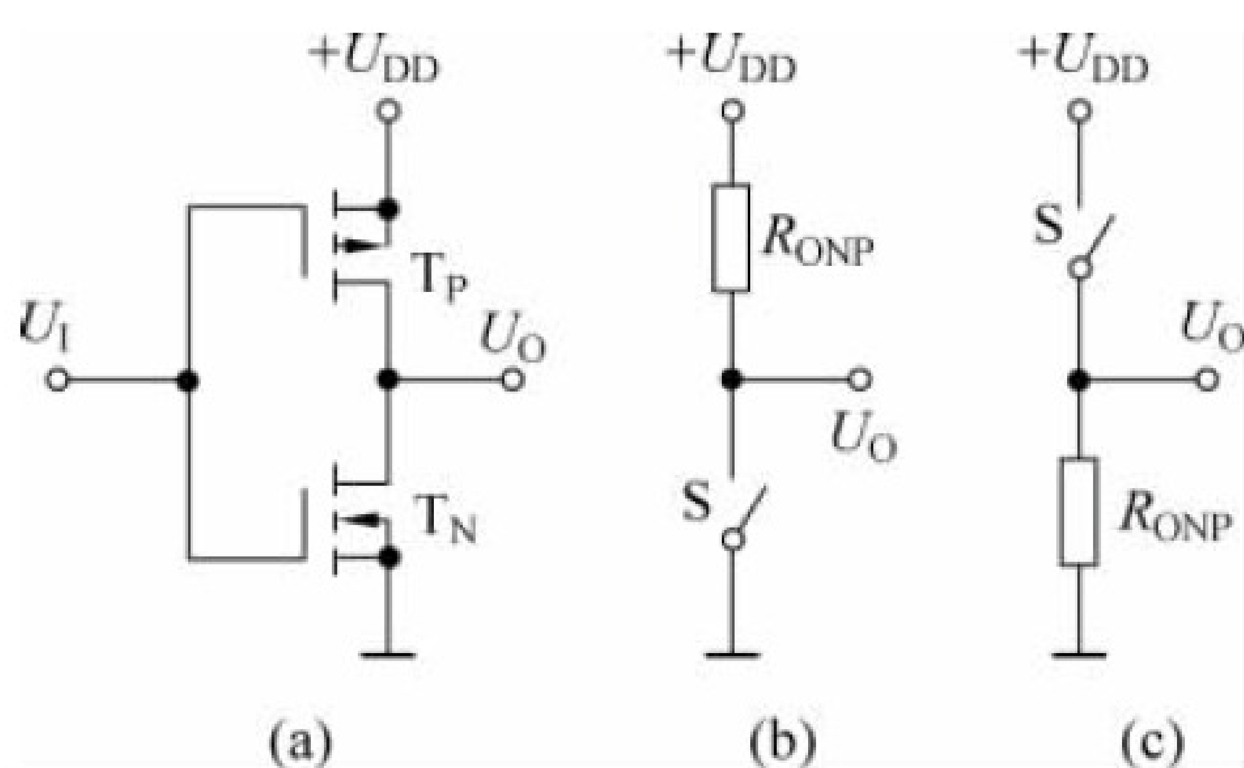


图 3.3.7　CMOS 反相器

当输入 $U_I=U_{IL}=0V$ 时，输入 T_N管的 $U_{GSN}=U_{IL}=0V$，小于 $U_{GS(TH)N}$，所以 T_N管截止。同时由于负载管 T_P的 $U_{GSP}=U_I-U_{DD}=0-U_{DD}=-U_{DD}$，小于 $U_{GS(TH)P}$，所以负载管 T_P导通，其简化等效电路如图 3.3.7(b)所示。输出电压 $U_O \approx U_{DD}$。

当输入 $U_I=U_{IH}=U_{DD}$时，输入 T_N管的 $U_{GSN}=U_{IH}$大于 $U_{GS(TH)N}$，所以 T_N管导通。同时由于负载管 T_P 的 $U_{GSP}=U_I-U_{DD}=U_{DD}-U_{DD}=0V$，大于 $U_{GS(TH)P}$，所以负载管 T_P截止，其简化等效电路如图 3.3.7(c)所示。输出电压 $U_O \approx 0V$。

故图 3.3.7(a)电路实现了反相的功能。

2. CMOS 传输门



如图 3.3.8 所示为 CMOS 传输门的电路图。它是由 P 沟道增强型 MOS 管和 N 沟道增强型 MOS 管并联互补组成。图 3.3.9 为它的逻辑符号。

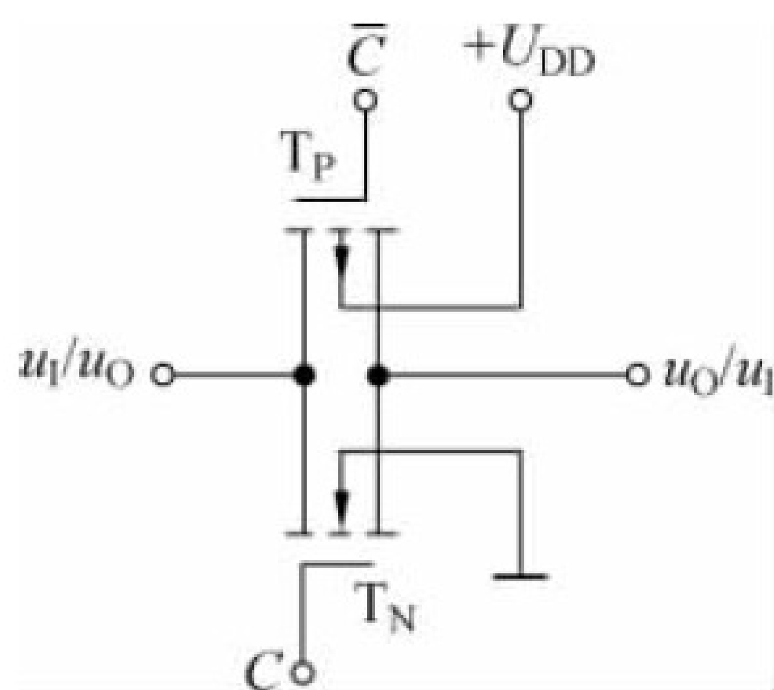


图 3.3.8　CMOS 传输门的电路图

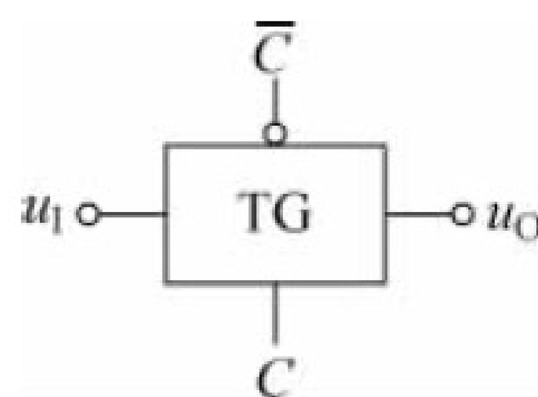


图 3.3.9　CMOS 传输门的逻辑符号

所谓并联互补，是指将 T_P管的源极和 T_N管的漏极相连，作为传输门的输入/输出端；将 T_P管的漏极和 T_N管的源极相连，作为传输门的输出/输入端。两个栅极受一对控制信号 C 和 $\overline{C}$ 控制。由于 MOS 管的结构是对称的，所以信号可以双向传输。U_I是被传输的模拟电压。

下面简要分析其工作原理：

(1) $C=0$，$\overline{C}=1$。由于 C 端为低电平 0V，$\overline{C}$ 端为高电平，则 T_N管、T_P管均截止，故传输

门截止。其输入和输出之间是断开的。

(2) $C=1$,$\overline{C}=0$。由于 C 端为高电平、$\overline{C}$ 端为低电平 0V,则 T_N管、T_P管均导通,故传输门导通。其输出 $U_O=U_I$。U_I可以是 0V 到 U_{DD}之间的任意电压值。

根据上面的分析,可见,CMOS 传输门类似于一个模拟开关,当有控制信号时,开关闭合,输出 $U_O=U_I$。无控制信号时,开关断开,不允许信号通过。

3. 其他功能 CMOS 门

如图 3.3.10 所示为 CMOS 与非门电路图,图中两个 PMOS 管(T_1、T_3)是并联起来的,两个 NMOS 管(T_2、T_4)是串联起来的。

当 A、B 两个输入端至少有一个为低电平 0 时,与该端相连的 NMOS 管截止而 PMOS 管导通。如两个 NMOS 管是串联的,则只要其中一个截止,输出端对地的电阻就非常大;如两个 PMOS 管是并联的,则只要其中一个导通,输出端和电源之间的电阻就很小。所以,输出为高电平 1。

只有当两个输入端 $A=B=1$ 时,两个 NMOS 管才都导通,两个 PMOS 管都截止。由于导通管的内阻很小,所以输出为低电平 0。即图 3.3.10 电路具有与非功能。

如图 3.3.11 所示为 CMOS 或非门电路图,图中两个 PMOS 管(T_1、T_3)是串联起来的,两个 NMOS 管是并联起来的。

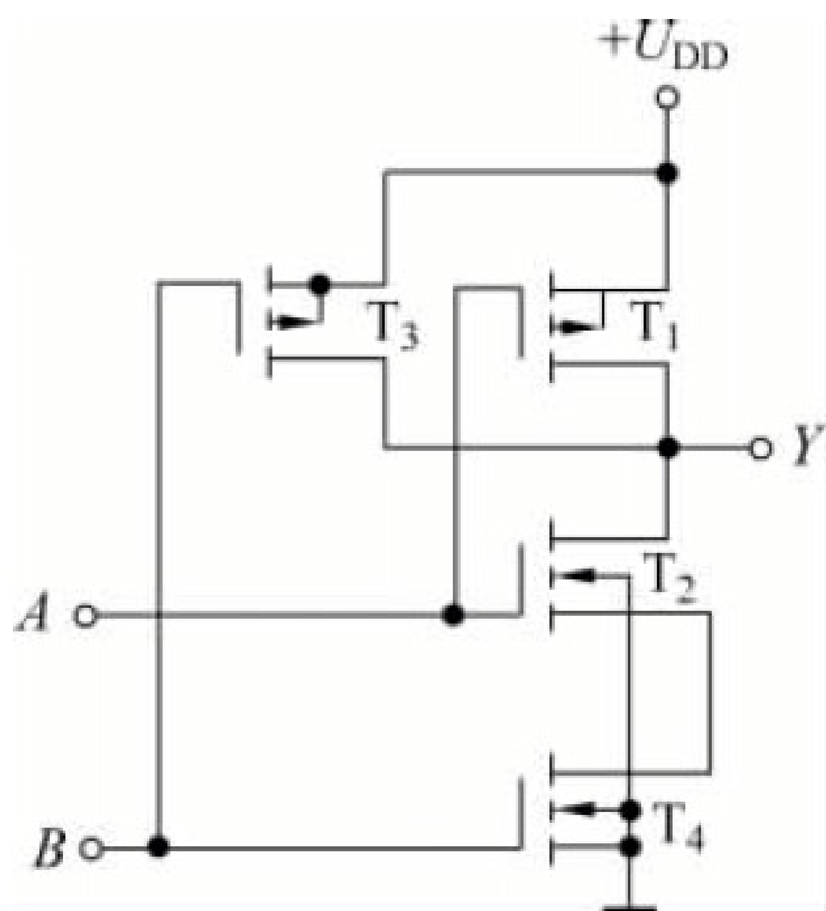


图 3.3.10 CMOS 与非门电路图

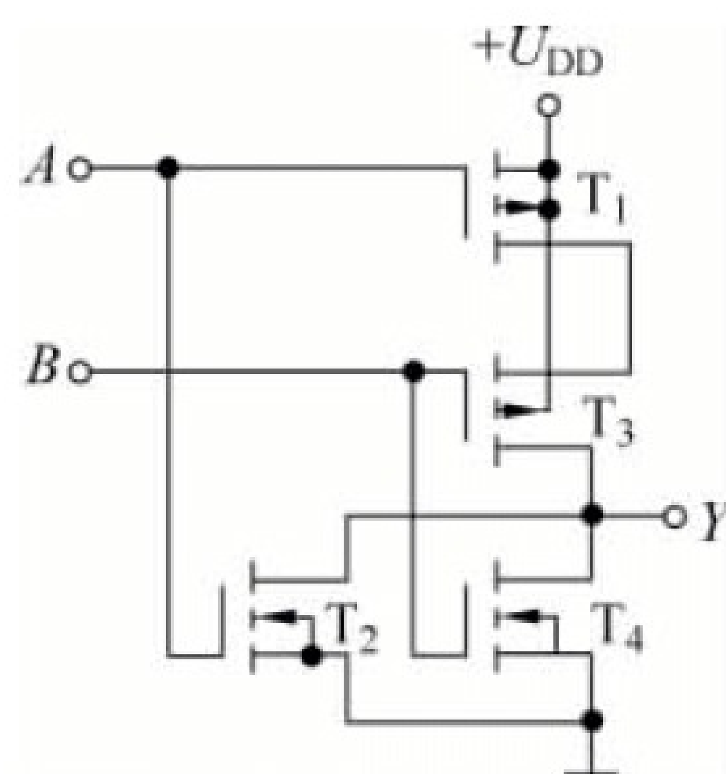


图 3.3.11 CMOS 或非门电路图

当 A、B 两个输入端至少有一个为高电平 1 时,与该端相连的 NMOS 管导通而 PMOS 管截止。因为两个 PMOS 管是串联的,则只要其中一个截止,输出端和电源之间的电阻就非常大;如果有一个 NMOS 管是导通的,则输出端和地之间的电阻就很小。所以,输出为低电平 0。

只有当两个输入端 $A=B=0$ 时,两个 PMOS 管才都导通,两个 NMOS 管都截止。由于导通管的内阻很小,所以输出为高电平 1。即图 3.3.11 电路具有或非功能。

3.3.3 用 CMOS 传输门和反相器构成逻辑电路

CMOS 传输门和反相器组合,可构成各种复杂的逻辑电路,可通过下面的例题来进一

步理解。

【例 3.3.1】 分析如图 3.3.12 所示电路的逻辑功能。

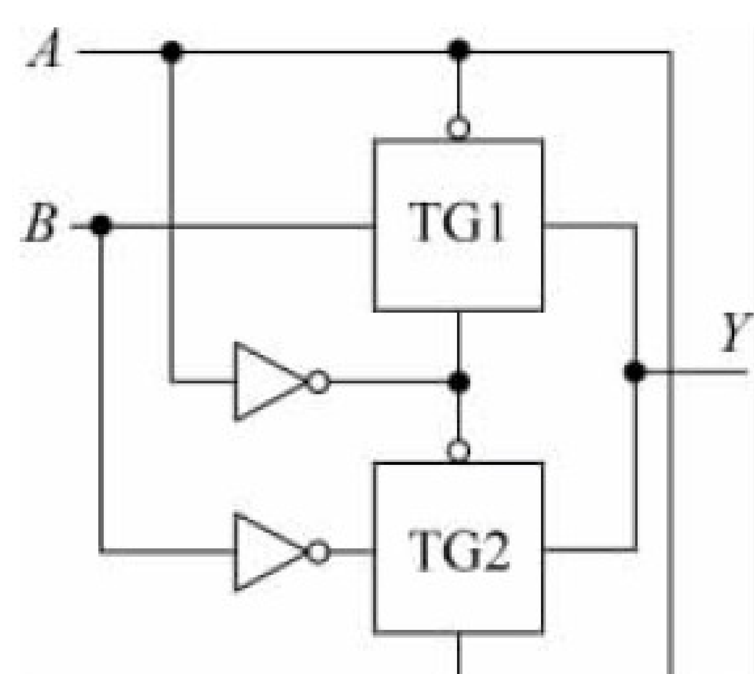


图 3.3.12 例 3.3.1 的图

解 (1) $A=0$。当 $A=0$ 时，根据传输门的控制特点，TG1 导通，TG2 截止，$Y=B$。

由 $A=0, Y=B$，有

$$Y=\overline{A}B$$

(2) $A=1$。当 $A=1$ 时，根据传输门的控制特点，TG2 导通，TG1 截止，$Y=\overline{B}$。

写成表达式，有

$$Y=A\overline{B}$$

因此，有

$$Y=A\overline{B}+\overline{A}B=A\oplus B$$

可见，图 3.3.12 所示电路为异或门。

【例 3.3.2】 分析如图 3.3.13 所示电路的逻辑功能。

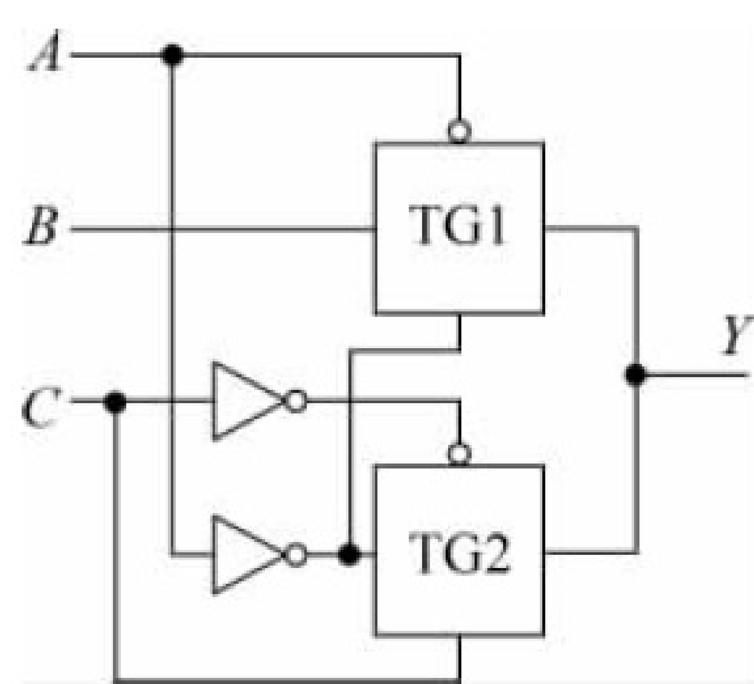


图 3.3.13 例 3.3.2 的图

解 (1) $A=0$。当 $A=0$ 时，根据传输门的控制特点，TG1 导通，TG2 截止，$Y=B$。

由 $A=0, Y=B$，有

$$Y=\overline{A}B$$

(2) $C=1$。当 $C=1$ 时，根据传输门的控制特点，TG2 导通，$Y=\overline{A}$。

写成表达式，有

$$Y=\overline{A}C$$

因此，有

$$Y=\overline{A}B+\overline{A}C$$

必须指出的是，上面的电路存在逻辑混乱的可能，即当 $A=0, C=1$ 时，TG1，TG2 同时导通，输出混乱，实际应用中应避免这种输入组合。

根据上面的例题，可总结分析方法如下：

(1) 逐个分析传输门导通条件及对应输出，写成与表达式。

(2) 将所有传输门输出或，求出最终输出。

(3) 分析所有传输门导通条件，若存在多个传输门同时导通的可能，则总结相关应用约束。

【例 3.3.3】 参考例 3.3.1、例 3.3.2 实现一个二输入或门。

解 设二输入或门输入变量为 A、B，输出为 Y。

为避免两个传输门同时导通的可能，可选择变量 A 为两个传输门控制变量。

由或门功能 $Y=A+B$，有

$$A=0,\quad Y=B$$
$$A=1,\quad Y=A$$

显然，上面是或门逻辑功能的全部描述，可根据上面的描述，参考图 3.3.12 所示电路画出电路，如图 3.3.14 所示。

当然，可选择变量 B 为 2 个传输门控制变量，有

$$B=0,\quad Y=A$$
$$B=1,\quad Y=B$$

读者可参考图 3.3.14 自己完成该设计。

复习与思考

电路如图 3.3.15 所示，写出该电路的输出 Y 的表达式。

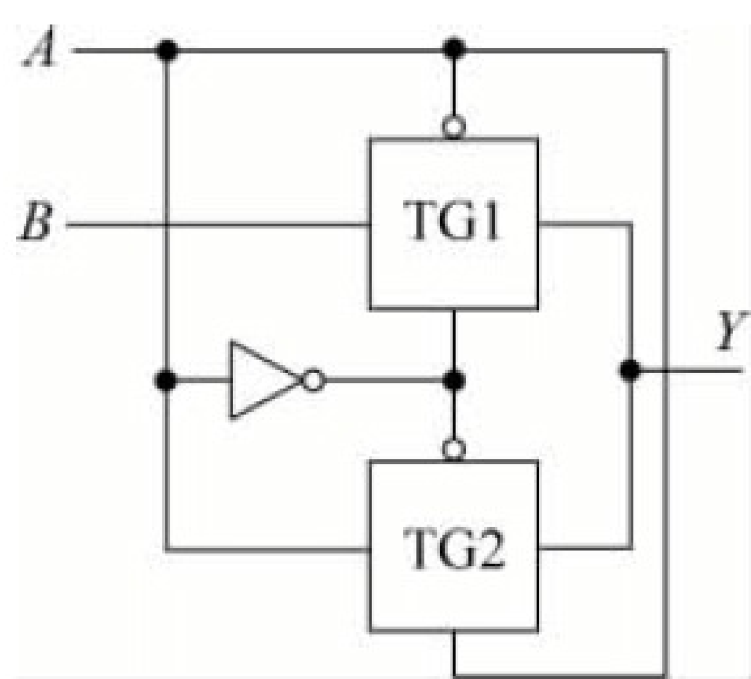


图 3.3.14　例 3.3.3 的图

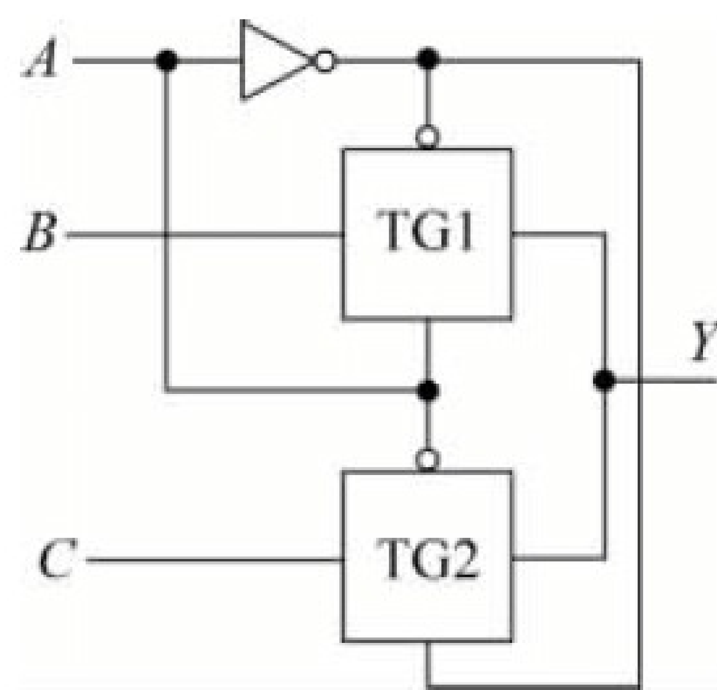


图 3.3.15　复习与思考的图

3.4　CMOS 集成逻辑门电路



CMOS 集成逻辑门电路是另一类流行的集成门电路。在很多场合下取代了 TTL 电路，已成为制造超大规模集成电路的主要器件。

3.4.1　集成 CMOS 反相器的电路特点

如图 3.3.7(a)所示的 CMOS 反相器为 CMOS 电路的基本结构形式。因为 MOS 管的栅极和衬底之间的绝缘介质非常薄，很容易被击穿(耐压约为 100V)。在集成的 CMOS 反相器中，增加了输入保护电路，以防止因接触到带静电电荷的物体时发生静电放电而损坏电路。

74HC 系列的 CMOS 器件中，多采用如图 3.4.1(a)所示的输入保护电路(虚线框中的电路)。图中，小椭圆框中的 C_1、C_2 为 T_1 管、T_2 管栅极寄生电容，右边椭圆框中为前面介绍的 CMOS 反相器电路。图中，D_1、D_2 等二极管导通压降 U_{DF} 约为 0.7V，反向击穿电压约为 30V。

显然，在输入信号的正常工作范围内($0<U_I<U_{DD}$)，输入保护电路中的 D_1、D_2 等二极管截止，不工作。当输入信号 U_I 大于 $U_{DF}+U_{DD}$ 时，D_1 管导通，将 T_1 管、T_2 管的栅极电位锁

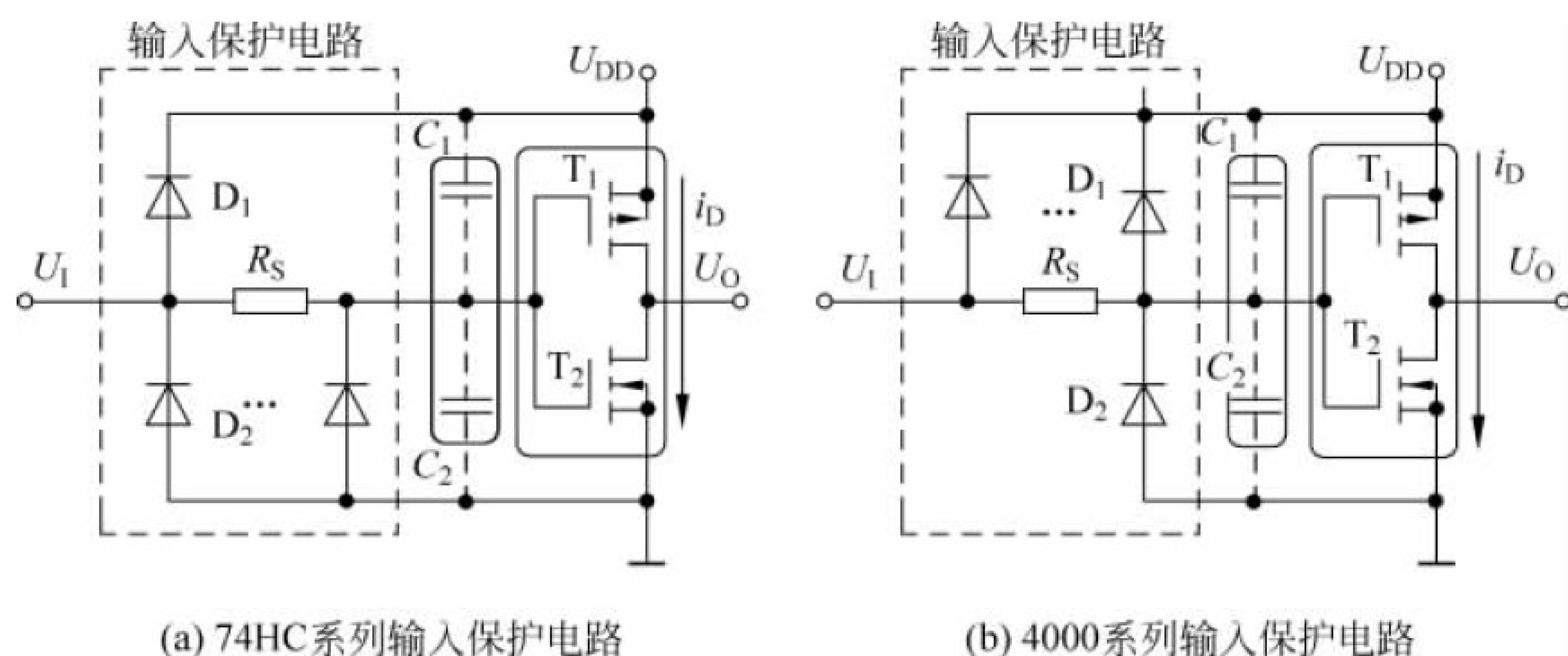


图 3.4.1 集成 CMOS 反相器

定在 $U_{DF}+U_{DD}$。当 U_I 小于 $-U_{DF}$ 时，D_2 管导通，将 T_1 管、T_2 管的栅极电位锁定在 $-U_{DF}$。可见，虚线框中的电路主要起输入保护的作用，正常情况下不起作用。

如图 3.4.1(b)所示的电路为 4000 系列的 CMOS 器件中常采用的输入保护电路。读者可自行分析该电路的工作原理。

必须指出的是，输入保护电路的保护措施是有一定限度的，D_1 管、D_2 管的正向导通电流过大或者反向电压过大均会损坏输入保护电路，进而损坏 MOS 管。因此，对 CMOS 器件，应特别注意器件的正确使用方法。

为帮助读者进一步理解 CMOS 反相器的特点，下面给出 CMOS 反相器的传输特性，如图 3.4.2 所示。

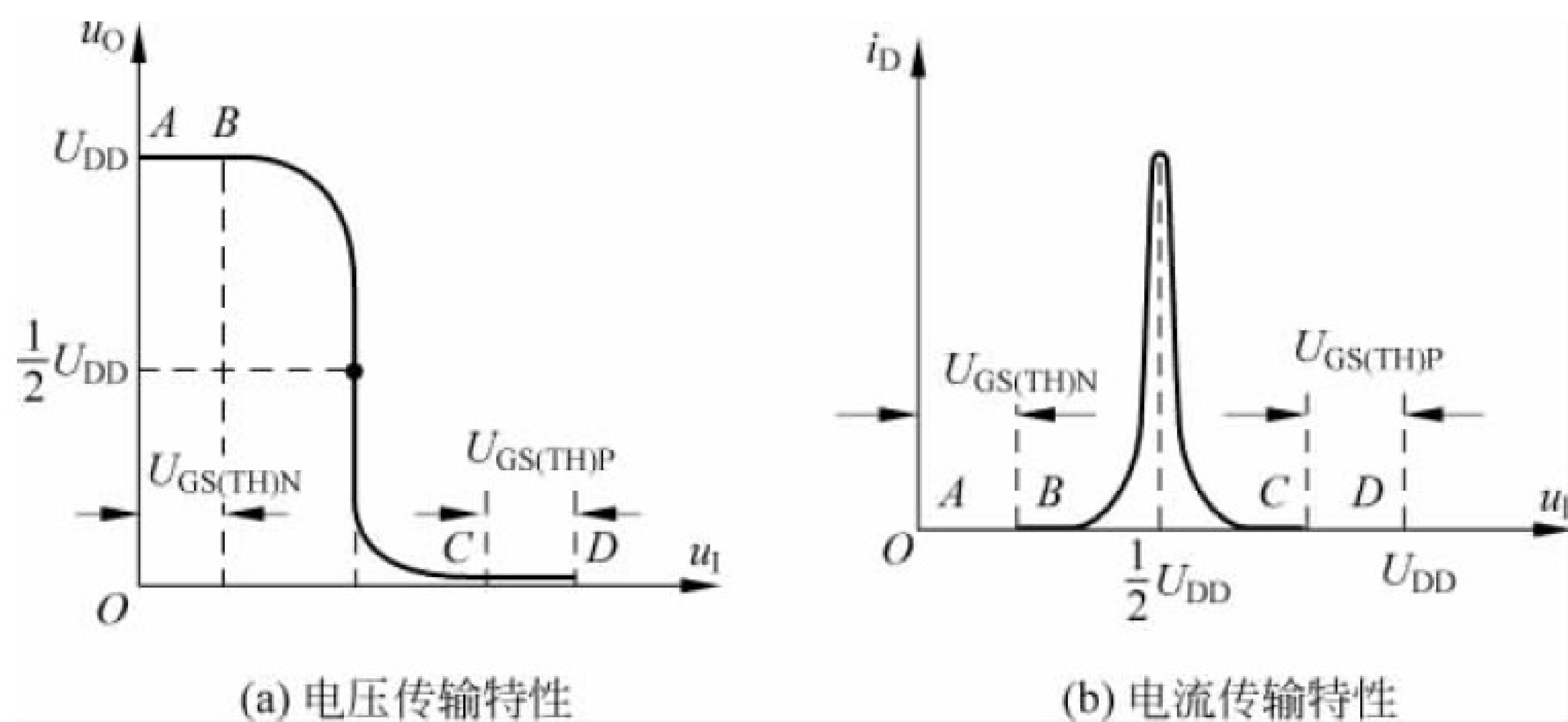


图 3.4.2 集成 CMOS 反相器传输特性

如图 3.4.2(a)所示为 CMOS 反相器的电压传输特性。当反相器工作在 AB 段的时候，$U_I<U_{GS(TH)N}$，上面的 MOS 管 T_1 导通，下面的 MOS 管 T_2 截止，输出 $U_O=1$（近似为 U_{DD}）。当反相器工作在 CD 段的时候，$u_I>U_{DD}-|U_{GS(TH)P}|$，上面的 MOS 管 T_1 截止，下面的 MOS 管 T_2 导通，输出 $U_O=0$（近似为 0）。BC 段为转折区，转折区中心点对应的输入电压称为 CMOS 反相器的阈值电压，用 U_{TH} 表示。

由图可看出，$U_{TH}=\frac{1}{2}U_{DD}$。

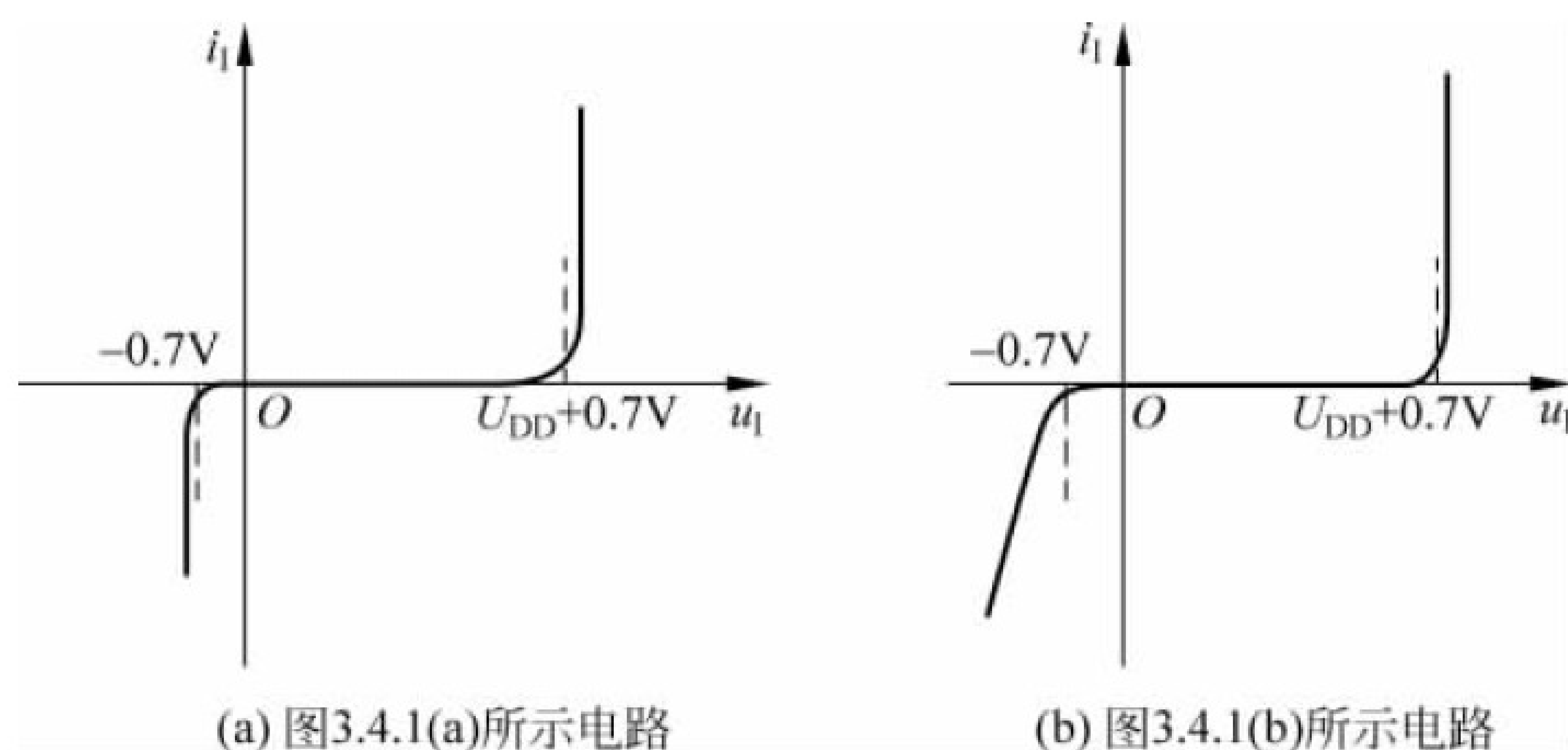


图 3.4.3　集成 CMOS 反相器的输入特性

由图 3.4.1 所示电路可知，当反相器工作在 AB 或 CD 段的时候，MOS 管 T_1、T_2总有一个截止，穿透电流 i_D为 0。只有当 $u_I=U_{TH}$时，MOS 管 T_1、T_2均导通，穿透电流 i_D较大。根据上述分析，可得到如图 3.4.2(b)所示的 CMOS 反相器的电流传输特性。

除传输特性外，还常使用输入、输出特性描述电路的特点。

电路的输入特性主要包括输入电压、输入电流、输入电阻等参数。对 CMOS 电路，输入高电平电压近似为 U_{DD}，输入低电平电压近似为 0。MOS 器件为电压控制器件，理想情况下，输入电阻无穷大，输入电流为 0，输入特性显而易见。考虑输入保护电路的作用，可总结如图 3.4.1 所示电路的输入特性，如图 3.4.3 所示。

电路的输出特性也常用输出电压、输出电流、输出电阻等参数来描述。对 CMOS 电路，输出高电平标准电压近似为 U_{DD}，输出低电平电压近似为 0。显然，随着输出电流绝对值的增大，输出高电平的电压将下降，输出低电平的电压将上升，可参考图 3.4.4 进一步理解集成 CMOS 反相器的输出特性。

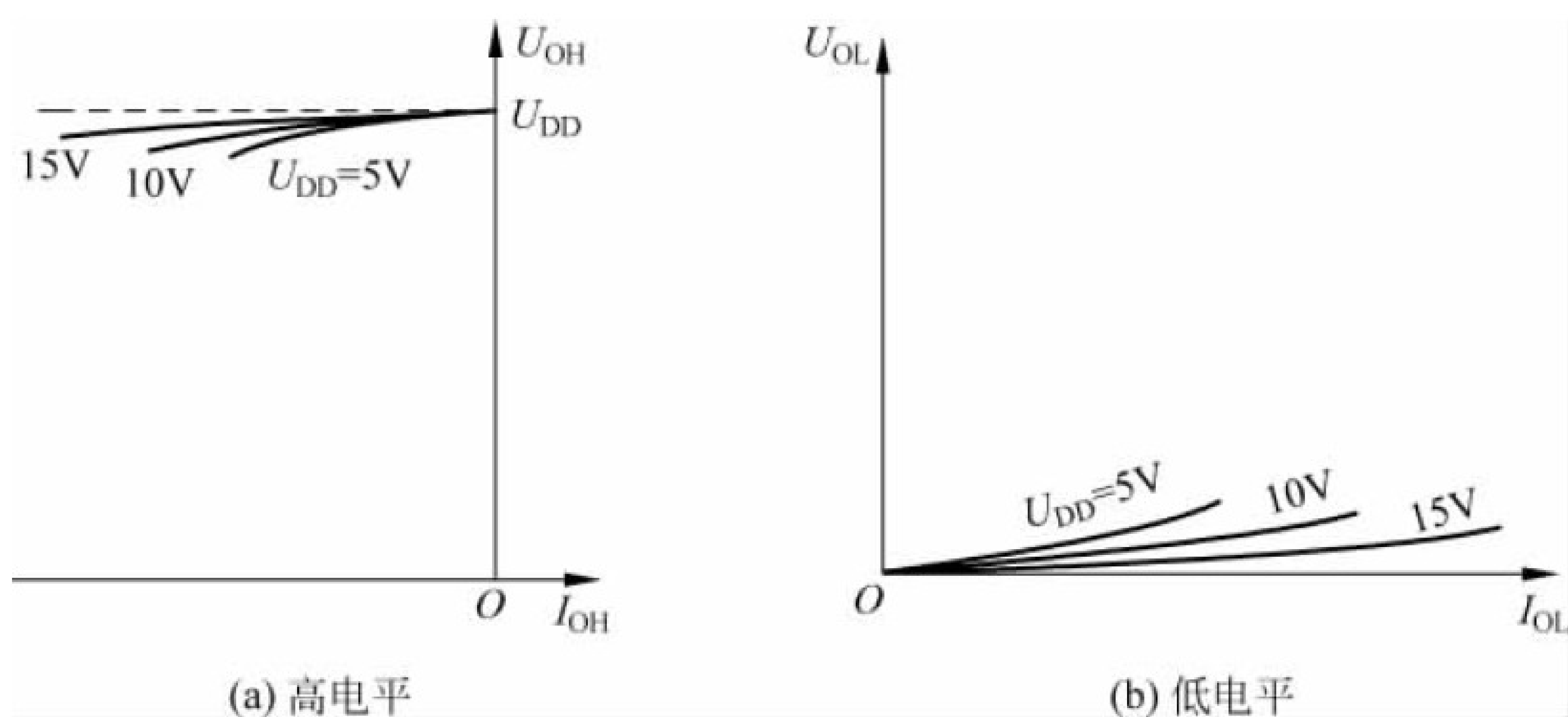


图 3.4.4　集成 CMOS 反相器的输出特性

必须指出的是，图 3.4.3 和图 3.4.4 所描述的输入、输出特性是 CMOS 反相器静态下的特性，CMOS 电路理论上输入电流为 0、功耗为 0 也是电路的静态特点。显然，CMOS 电路高、低电平两种状态的转换存在一个过渡过程，参考如图 3.4.2(b)所示的 CMOS 反相器的电流传输特性可知，从动态角度，CMOS 电路功耗并不为 0，存在着一定的动态功耗，但是

与 TTL 电路相比，功耗小得多。

基于上面的分析，可归纳 CMOS 反相器的主要优点如下：

(1) 静态功耗极低。由于 CMOS 电路中的 T_1、T_2两管不是同时导通，而截止管的电阻又很高，这就使得在任何时候流过电路的电流都很小，仅为管子的漏电流(小于微安级)，所以这种互补的非门(反相器)电路的功耗很小，各个门的静态功耗只有 0.01mW(而 TTL 每个门的功耗约 10mW)。

(2) 抗干扰能力较强。由于 CMOS 电路的阈值电平近似等于 $1/2U_{DD}$，在输入信号变化时，过渡变化陡峭，所以低电平噪声容限和高电平噪声容限近似相等，而且随着电源电压的增加，抗干扰能力也增强。

(3) 输出电压 U_O 的逻辑摆幅较大。由于输出低电平约为 0V，输出高电平约为 U_{DD}，所以输出电压的逻辑摆幅较大。

当然，任何系统有某个方面的突出优点，必存在某些方面相对应的不足。CMOS 电路理论上输入电流为 0、功耗为 0，这是其突出优点。输出高电平标准电压近似为 U_{DD}，输出低电平电压近似为 0，高低电平幅度大，有利于提高电路的可靠性。输入电流近似为 0，使 CMOS 门电路的驱动能力与 TTL 电路相比大幅提升。直流状态下，CMOS 门电路的扇出数是非常大的。低频(小于 1MHz)工作状态下，CMOS 电路的扇出数一般也可达 50 以上。

TTL 电路的高电平标准电压一般为 3.2V，与 TTL 电路相比，CMOS 电路的高电平近似为 U_{DD}。因为 CMOS 电路的高电平比 TTL 电路的高电平高得多，因此，同等条件下，CMOS 电路高、低电平两种状态的转换过渡需要更长的时间，CMOS 门电路的传输延时也相对要大一些。

当然，随着集成电路制作工艺水平的提高，当今的 CMOS 集成门电路传输延时也能达到很小的水平，如 74AHC04 平均传输延时只有 5ns。基于这些特点，CMOS 电路已经在很多领域取代了 TTL 电路，应用十分广泛。

3.4.2 CMOS 集成电路的正确使用

1. 输入电路的静电防护

CMOS 集成电路的输入端一般均设置了类似图 3.4.1 所示的输入保护电路。该保护电路中二极管及限流电阻的几何尺寸有限，只能承受有限的静电电压。CMOS 集成电路使用过程中难免接触到带静电的物体，其中的某些物体极易产生高压。如果工作人员穿的是毛衣织物之类的衣裤，则有时会因摩擦产生高达上千伏的静电，如果这个静电电压被加到 CMOS 电路输入端，将对 CMOS 电路造成严重危害。

应用实践中，应设法避免静电对 CMOS 电路造成危害，主要措施有：

(1) 储存运输环节中，将器件插在泡沫塑料上并采用金属屏蔽包装，不要使用容易产生静电高压的化工材料和化纤织物包装。从包装中取出器件时应避免用手直接触摸器件的

引脚，将器件放在接地的导电平面上。

(2) 断电插拔器件及电路板。

(3) 组装、调试维护环节无静电作业，使相关的工作台良好接地，工作人员的服装、手套等选用无静电的原料制作。

(4) 不用的输入端不应悬空，可根据逻辑功能要求接0或1。

2. 输入电路的过流保护

输入保护电路中的钳位二极管电流容量有限，一般为1mA，因此，在可能出现较大输入电流的场合应采用防护措施。例如，在信号源和输入端之间串接保护电阻；输入端接有大电容时，在输入端和电容之间接入保护电阻等。

3. CMOS电路的锁定效应及其防护

CMOS电路由于输入太大的电流，内部的电流将急剧增大。除非切断电源，内部的电流将一直增大，直到饱和，这种现象称为锁定效应。当产生锁定效应时，CMOS的内部电流能达到40mA以上，很容易烧毁芯片。

主要防御措施有：

(1) 在输入端和输出端加钳位电路，使输入和输出不超过规定电压。

(2) 芯片的电源输入端加去耦电路，防止电源端出现瞬间的高压。

(3) 当系统由几个电源分别供电时，按下列顺序开启开关：开启时，先开启CMOS电路的电源，再开启输入信号和负载的电源；关闭时，先关闭输入信号和负载的电源，再关闭CMOS电路的电源。

此外，与TTL类似，CMOS系列门电路也有漏极开路输出(OD门)门电路、三态输出门电路等。读者可参考TTL系列的类似输出结构的门电路使用方法理解这些门电路的使用。

3.4.3 CMOS集成电路与TTL电路的混合使用



目前，TTL系列和CMOS系列两大电路并存，在有些场合下，经常涉及两种器件相互连接的问题。

对照CMOS电路、TTL电路特点，从连接角度，二者主要有以下差异：

CMOS电路高电平约为U_{DD}；TTL电路高电平理论最大值3.6V。TTL电路输入电流为−1mA，CMOS电路输入电流为0mA，输出电流也小，74HC/74HCT系列的I_{OHMAX}、I_{OLMAX}均为4mA。

基于上面的差异，当用TTL驱动CMOS电路时，读者不难发现，TTL电路的高电平比CMOS电路高电平小得多。查集成电路手册，所有TTL系列的U_{OHMIN}均比74HC和74AHC等CMOS系列电路的U_{IHMIN}小，因此，当用TTL驱动CMOS电路时，必须设法提

高 TTL 电路输出高电平的下限值。

一种简单的解决方法是在 TTL 的输出端与电源之间接入上拉电阻 R_L，参考电路如图 3.4.5 所示。

在如图 3.4.5 所示电路中，当 TTL 输出为高电平时，TTL 输出管处于截止状态。TTL 输出级截止时的漏电流很小，CMOS 输入电流近似为 0，R_L 上的压降近似为 0，因此，只要 R_L 的阻值不是特别大，输出高电平就会被提升为 U_{DD}。

关于 R_L 的取值范围，有兴趣的读者可参考例 3.2.3 进一步理解。

当用 CMOS 驱动 TTL 电路时，尽管 CMOS 输出电流较小，但主流的 CMOS 系列电路依然可以直接驱动任何系列的 TTL 电路，但是可以驱动负载门的个数较少。当负载门较多，要求负载电流较大时，可使用电流放大器扩大输出电流，参考电路如图 3.4.6 所示。

必须指出的是，如图 3.4.6 所示的电流放大器具有反相器的逻辑功能，可在原 CMOS 门电路的基础上再接一个 CMOS 反相器后接电流放大器，以确保电路的逻辑功能没有改变。

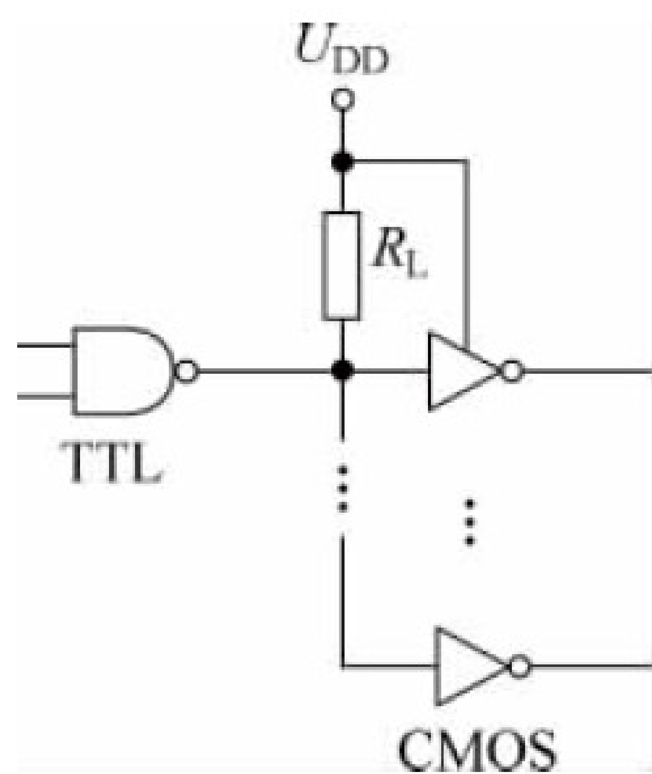


图 3.4.5　TTL 驱动 CMOS

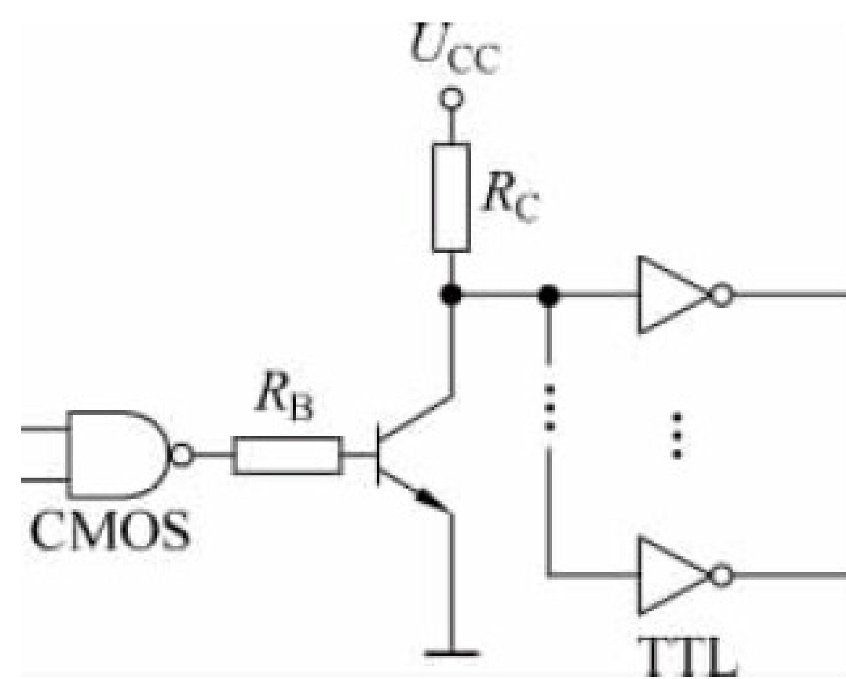


图 3.4.6　CMOS 驱动 TTL

3.4.4　CMOS 数字集成电路的各种系列

CMOS 电路的成功研制已有半个多世纪，伴随着 CMOS 电路制作工艺的不断改进，CMOS 电路的性能得到了迅速提高，各国的半导体器件制作商先后推出了多种系列的 CMOS 数字集成电路。

下面主要以 TI 公司生产的 CMOS 系列电路为主介绍不同产品系列的特点。

1. 4000/14000 系列

该系列为最早投向市场的 CMOS 产品。4000 系列由美国 RCA 公司生产，14000 系列为美国 Motorola 公司的产品。该系列产品的最大特点是工作电源电压范围宽（3～18V）、功耗小、价格低廉；主要不足是传输延时很长，约为 100ns。此外，该芯片的负载能力也很弱，当工作电源为 5V 时，输出为高电平时输出的最大负载电流和输出为低电平时输出的最

大负载电流均只有 0.5mA。

基于上面的不足，该系列电路已逐渐被后来出现的 HC/HCT 系列取代。

2. 74HC/HCT 系列

74HC(High-speed CMOS)/HCT(High-speed CMOS，TTL Compatible)系列是 TI 公司生产的高速 CMOS 产品系列。该系列产品的推出及初期推广应用便以取代 TTL 电路作为一个重要目标。从产品性能角度，CMOS 电路功耗上远优于 TTL 电路。此外，74HC/HCT 系列通过一系列改进，传输延时已缩短到 10ns，负载能力提高到 4mA，可以和 TTL 的 74LS 系列匹敌。

74HCxxx、74HCTxxx 是 74LSxxx 同序号的翻版。型号最后几位数字相同，表示电路的逻辑功能、引脚排列完全兼容，为用 74HC/HCT 系列替代 74LS 提供了方便。74HC 系列和 74HCT 系列在传输延时、负载能力方面基本相同，主要在工作电压及输入信号电平方面有所不同。74HC 系列可在 2～6V 间的任何电源电压下工作，工作电平与 TTL 电平不匹配。74HCT 系列为 TTL 兼容系列，工作电压、工作电平与 TTL 完全兼容，可与 TTL 系列芯片混合使用。

3. 74AHC/AHCT 系列

74AHC(Advanced High-speed CMOS)/AHCT(Advanced High-speed CMOS，TTL Compatible)系列是 TI 公司生产的改进的高速 CMOS 产品系列。通过一系列改进，芯片的工作速度提高了近 1 倍，负载能力提高到 8mA。该系列产品与 74HC/HCT 系列产品保持了高度兼容，非常有利于老产品的升级与更新，是目前应用最广泛的 CMOS 器件之一。

4. 74LVC/ALVC/74AVC 系列

74LVC(Low-Voltage CMOS)/ALVC(Advanced Low-Voltage CMOS)/AVC(Advanced Very-Low-Voltage CMOS)系列是 TI 公司生产的低电压 CMOS 逻辑系列产品。该系列产品特点是工作电压低，传输延时短且负载电流大。74LVC/ALVC 系列工作电压为 1.65～3.3V，负载电流可达 24mA。74AVC 系列工作电压为 1.2～3.6V。

从速度、功耗、工作电压等综合参数角度，该系列是目前 CMOS 系列中性能最好的产品系列，能满足高性能数字系统设计的需要，广泛应用于移动式便携电子设备，如手机、笔记本、数码相机等。当然，该系列产品工作电压低、逻辑电平摆幅小，因此抗干扰能力弱，在电磁干扰较大的工业环境中应用有些不足。

TI 公司生产的不同 CMOS 系列性能比较表如表 3.4.1 所示(以反相器为例)，读者可

参考 TI 公司产品的性能参数理解其他半导体器件公司的类似产品及其之间的差异。

表 3.4.1　CMOS 系列器件(74××00)主要性能比较

参数名称与符号	系　　列					
	74HC	74HCT	74AHC	74AHCT	74LVC	74ALVC
U_{ILMAX}/V	1.35	0.8	1.35	0.8	0.8	0.8
U_{OLMAX}/V	0.33	0.33	0.44	0.44	0.55	0.55
U_{IHMIN}/V	3.15	2	3.15	2	2	2
U_{OHMIN}/V	4.4	4.4	4.4	4.4	2.2	2.0
I_{ILMAX}/μA	−1	−1	−1	−1	−5	−5
I_{OLMAX}/mA	4	4	8	8	24	24
I_{IHMAX}/μA	1	1	1	1	5	5
I_{OHMAX}/mA	−4	−4	−8	−8	−24	−24
t_{pd}/ns	9	14	5.3	5.5	3.8	2
功耗电容/pF	20	20	12	14	8	23
工作电压/V	2～6	4.5～5.5	2～5.5	4.5～5.5	1.65～3.6	1.2～3.6

注：1. 74LVC、74ALVC 系列给出的参数(工作电压除外)是 3V 工作电压下的参数，其他系列给出的参数是 4.5V 工作电压下的参数；

2. U_{OLMAX}、U_{OHMIN} 是表中给出的最大负载电流下的输出电压。

除 TTL、CMOS 两大系列外，TI 公司还研制推出了 Bi-CMOS 电路。Bi-CMOS 电路是双极型 CMOS(Bipolar-CMOS)电路的简称。这种电路的结构是逻辑部分采用 CMOS 结构，输出级采用双极型三极管结构。这种电路兼有 CMOS 电路低功耗、高集成度和 TTL 电路高驱动能力的优点，常用于需要输出大驱动电流的场合，如计算机的总线接口、数字系统输出缓冲器、驱动器和锁存器等。74 系列的 Bi-CMOS 电路主要有 74ABT、74ALB、74ALVT 等产品系列，有兴趣的读者请参考相关书籍。

复习与思考

根据本节知识解释智能手机待机状态下比工作状态下更省电的原因。

3.5　利用硬件描述语言描述常用逻辑运算

硬件描述语言(Hardware Description Language，HDL)是电子系统硬件行为描述、结构描述、数据流描述的语言。采用硬件描述的方法设计数字电路是目前流行的电路设计方法。

3.5.1　硬件描述语言简介



门电路是构成数字电路的基础，逻辑运算功能最终是通过电路实现的。具体实现上，可利用 CMOS 或 TTL 门电路实现各种数字电路。

当然，对于特定的数字系统，有经验的设计师总是会根据应用系统要求，合理划分构成

系统的电路模块,如电源模块、输入模块、显示模块、控制模块等。即单独设计各模块电路,画出各模块电路图,分块调试各模块;之后,将各模块电路焊接、连接、组装成整体,整体调试测试其功能并完成最终系统的设计与调试。

这种方法是传统硬件系统设计的一般方法,至今仍在广泛应用。当然,这种方法的不足也很明显,如不方便修改其功能、不利于后期修改维护,电路性能与焊接、组成工艺紧密相关。当然,为了改进电路性能与焊接、组成工艺紧密相关的不足,可以考虑将设计的电路模块做成一个专用集成电路(ASIC),以提高设计电路的应用性能。

集成电路问世已经超过半个世纪。伴随着集成度的提高,单个芯片便已成为功能强大的系统,且可编程。硬件设计越来越倾向于与系统设计和软件设计结合,各半导体器件厂商纷纷推出了自己的硬件描述语言。

利用硬件描述语言,数字电路系统的设计可以从顶层到底层(从抽象到具体)逐层描述自己的设计思想,用一系列分层次的模块来表示极其复杂的数字系统。然后,利用电子设计自动化(EDA)工具,逐层进行仿真验证,再把其中需要变为实际电路的模块组合,经过自动综合工具转换到门级电路网表。最后,用专用集成电路 ASIC 或现场可编程门阵列 FPGA 自动布局布线工具,把网表转换为要实现的具体电路。

硬件描述语言是高级语言,接近人的思维习惯,电路设计也由传统的如何设计演变为设计什么。这种高层次的方法已被广泛采用。据统计,目前在美国硅谷约有 90%以上的 ASIC 和 FPGA 采用硬件描述语言进行设计。

硬件描述语言发展至今已有数十年的历史,并成功地应用于设计的各个阶段,如建模、仿真、验证和综合等。到 20 世纪 80 年代,已出现了上百种硬件描述语言,对设计自动化起到了极大的促进和推动作用。得到普遍认同的有 VHDL 和 Verilog HDL。

VHDL(Very-High-Speed Integrated Circuit Hardware Description Language)诞生于 1982 年,1987 年成为国际标准。Verilog HDL 是由 Gateway 设计自动化公司的工程师于 1983 年末创立的,是另一种流行的硬件描述语言,以文本形式来描述数字系统硬件的结构和行为。Verilog HDL 1995 年成为国际标准,之后进行了多次更新,最主流版本为 Verilog-2001。本书主要介绍 Verilog HDL 硬件描述语言。

3.5.2 Verilog HDL 语言的基本结构



当采用硬件描述语言描述一个复杂的硬件电路时,设计人员总是将复杂的功能划分为若干简单的功能,每个简单功能对应一个模块(module)。设计人员可以采取“自顶向下”的思路,将复杂的功能模块划分为低层次的模块。这一步通常是由系统级的总设计师完成,而低层次的模块则由下一级的设计人员完成。自顶向下的设计方式有利于系统级别层次划分和管理,并提高了效率、降低了成本。

基于上面的思路,Verilog HDL 采用基于 C 语言的语法以模块为基础进行硬件系统的设计。

使用 Verilog 描述硬件的基本设计单元是模块。构建复杂的电子电路,主要通过模块的相互连接调用来实现。Verilog 中的模块类似 C 语言中的函数,它能够提供输入、输出端口,可以实例调用其他模块,也可以被其他模块实例调用。模块中可以包括组合逻辑部分、过程时序部分。

【例 3.5.1】 分析下面代码的结构及其描述的逻辑功能。

```
module adder(c,s,a,b,cin);      /* 模块名 */
      input [2:0] a,b;          /* 描述组输入 */
      input cin;                /* 描述输入 */
      output c;                 /* 描述输出 */
      output [2:0] s;           /* 描述组输出 */
      assign {c,s} = a + b + cin;  /* 描述功能 */
endmodule
```

解 (1) 代码的结构分析。硬件描述语言是高级语言,可按照日常的思维习惯去理解这些代码。Verilog 的基本单元是模块(module),上面的代码被包含在关键字 module、endmodule 之内,可见从关键字 module 开始到 endmodule 之间的代码均为模块的具体代码。

Verilog 遵循 C 语言的语法,代码中的“(…)”为 C 语言的函数标识,因此,adder 为模块名(函数名)。

电路应具有必须的输入及输出,input、output 具体描述了电路所具有的输入及输出。

基于上面的分析,可总结 Verilog 模块的结构如下:

```
module <模块名>(输入输出表)
    <输入输出定义>
    < 模块条目>
endmodule
```

其中,input、output 为输入输出描述关键词。例如,“input [2:0] a;”描述了一组输入“a2、a1、a0”。

(2) 逻辑功能分析。

电路的输入:{a2、a1、a0}、{b2、b1、b0}、cin

电路的输出:{s2、s1、s0}、c

只有一条功能描述语句“assign {c,s}=a+b+cin;”似乎执行了数据 a、b、cin 的加法并将结果送给了{c,s}。

(3) 逻辑功能验证(具体实现过程见附录 C)。逻辑功能最终是通过各种门电路相互连接实现的。可利用开发工具将硬件描述语言的代码转换为实现描述功能的逻辑门级电路的连接网表,之后,直接选用 CPLD/FPGA 实现最终输出的门级网表。各硬件厂商均基于自己的 CPLD/FPGA 开发了仿真分析工具软件,可将上面的代码输入 Quartus Ⅱ 中,编译代码,设置相关的输入,可求出如图 3.5.1 所示的仿真波形(Quartus Ⅱ 9.0 环境下的仿真波形)。

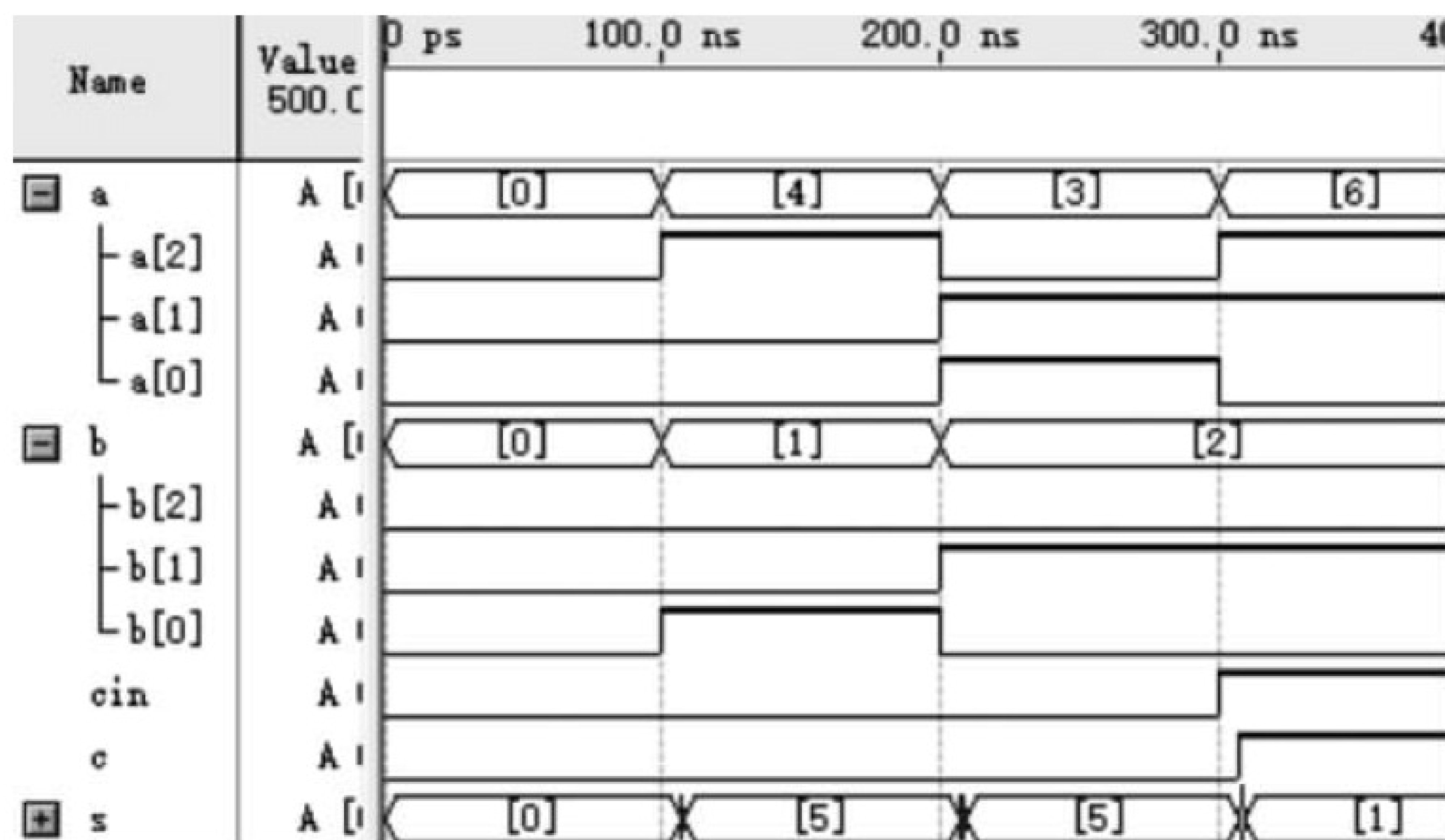


图 3.5.1　例 3.5.1 的仿真波形

仿真结果显示该代码功能为 3 位二进制加法器。其中,{a2、a1、a0}、{b2、b1、b0}为两个相加的 3 位二进制,cin 为来自低位的进位。{s2、s1、s0}为结果,c 为向高位的进位。如最后一个单元,6+2+进位,结果为 1,向高位的进位为 1。

assign 语句含义:连续赋值语句(assign 语句),主要用于对 wire 型(线型)变量的赋值。

显然,对电路而言,上面的{a2、a1、a0}等变量对应具体的外部输入接线端,Verilog 中把这类变量称为 wire 型(线型)变量(默认的变量类型)。根据该电路的特点,只有 a、b、cin 等输入连续保持,才能获得稳定的连续输出 s。a、b、cin 的值一旦发生改变,将立即反映到输出 s 上,这便是 assign 语句中连续赋值的含义。

基于上面的分析,可初步总结 Verilog HDL 代码编写方法:

(1) 编写一个由 module … endmodule 组成的包含名字的空模块(Quartus Ⅱ 中模块名必须与文件名相同!)。

(2) 根据逻辑功能定义电路的输入及输出,各变量名应在模块名后的()中注明。使用 input、output 具体描述电路的输入输出。

(3) 根据逻辑功能,使用 assign 语句给出输出结果。

3.5.3　利用 Verilog HDL 描述常用逻辑运算



下面通过两个实例介绍如何用 Verilog HDL 描述常用逻辑运算。

【例 3.5.2】 请对照 3 位二进制加法器代码描述一个二输入或门。

解　(1) 先定义模块名。定义模块名为 ort(注意,不可定义为 or2,因为 or2 为 Quartus Ⅱ 系统定义的二输入或门的库元件名)。

(2) 定义电路的输入及输出。根据二输入或逻辑的功能,可定义两个输入 a、b,一个输出 y。可写出初步代码如下:

```
module ort(a,b,y);
```

```
input   a,b;   output   y;
endmodule
```

(3) 根据逻辑功能,使用 assign 语句给出输出结果。或逻辑功能吻合连续赋值的逻辑要求,可使用 assign 语句给出输出结果。Verilog 遵循 C 语言的语法,C 语言位或运算符为"|",具体语句如下:

```
assign y = a|b;
```

(4) 仿真验证。仿真波形如图 3.5.2 所示,仿真结果显示该代码功能为二输入或门。

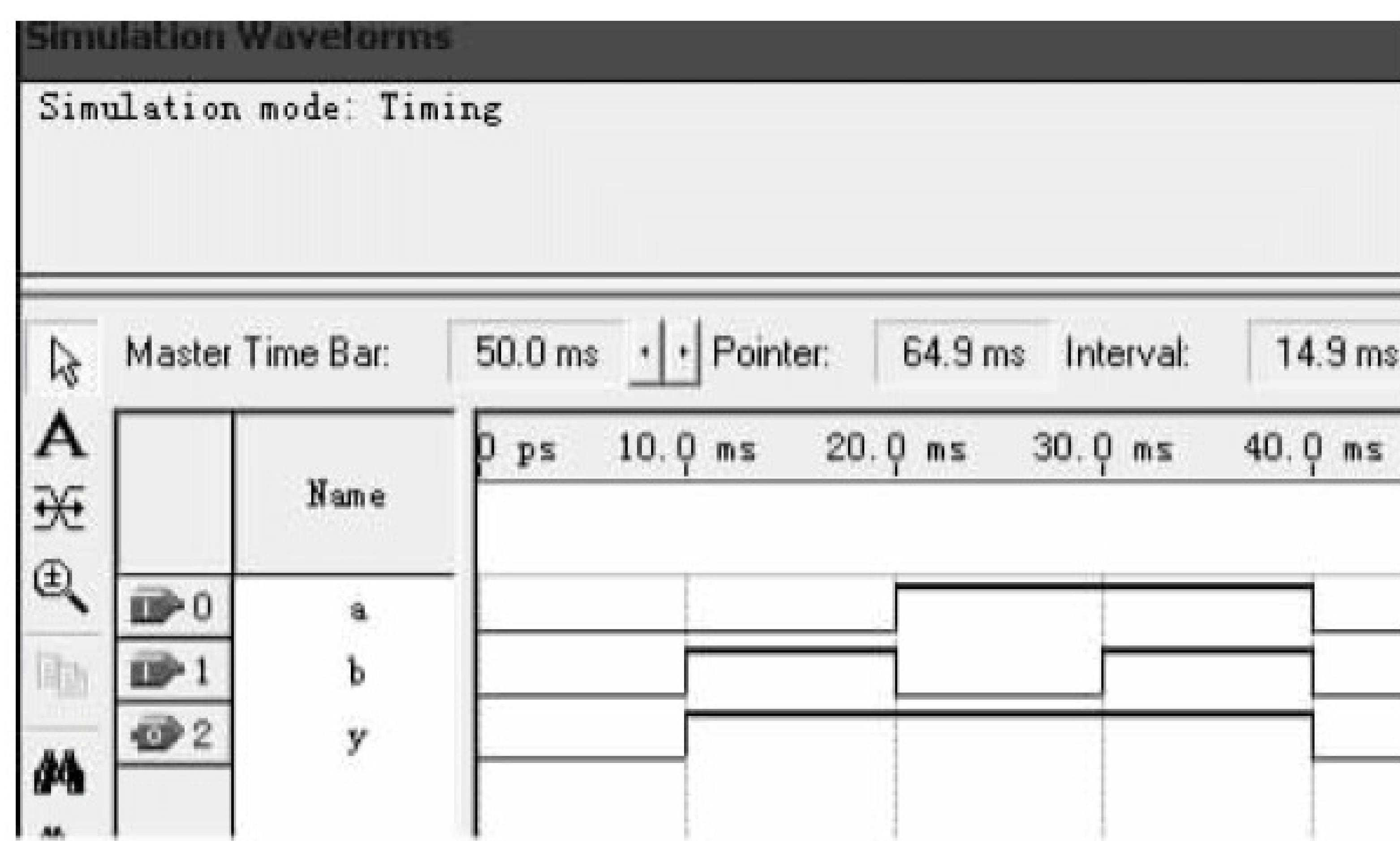


图 3.5.2　例 3.5.2 的仿真波形

【例 3.5.3】 参考二输入或门代码描述一个二输入与非门。

解 (1) 先定义模块名。定义模块名为 nandt(注意,不可定义为 nand2,因为 nand2 为 Quartus Ⅱ系统定义的二输入与非门的库元件名)。

(2) 定义电路的输入及输出。根据二输入与非门的功能,可定义两个输入 a、b,一个输出 y。可写出初步代码如下:

```
module nandt(a,b,y);
input   a,b;   output   y;
endmodule
```

(3) 根据逻辑功能,使用 assign 语句给出输出结果。与非逻辑功能吻合连续赋值的逻辑要求,可使用 assign 语句给出输出结果。Verilog 遵循 C 语言的语法,C 语言位与运算符为"&",非运算符"~",具体语句如下:

```
assign y = ~(a&b);
```

(4) 仿真验证。仿真波形如图 3.5.3 所示,仿真结果显示该代码功能为二输入与非门。

复习与思考

根据连续赋值的含义给出一种不适合用 assign 语句给出输出结果的应用实例。

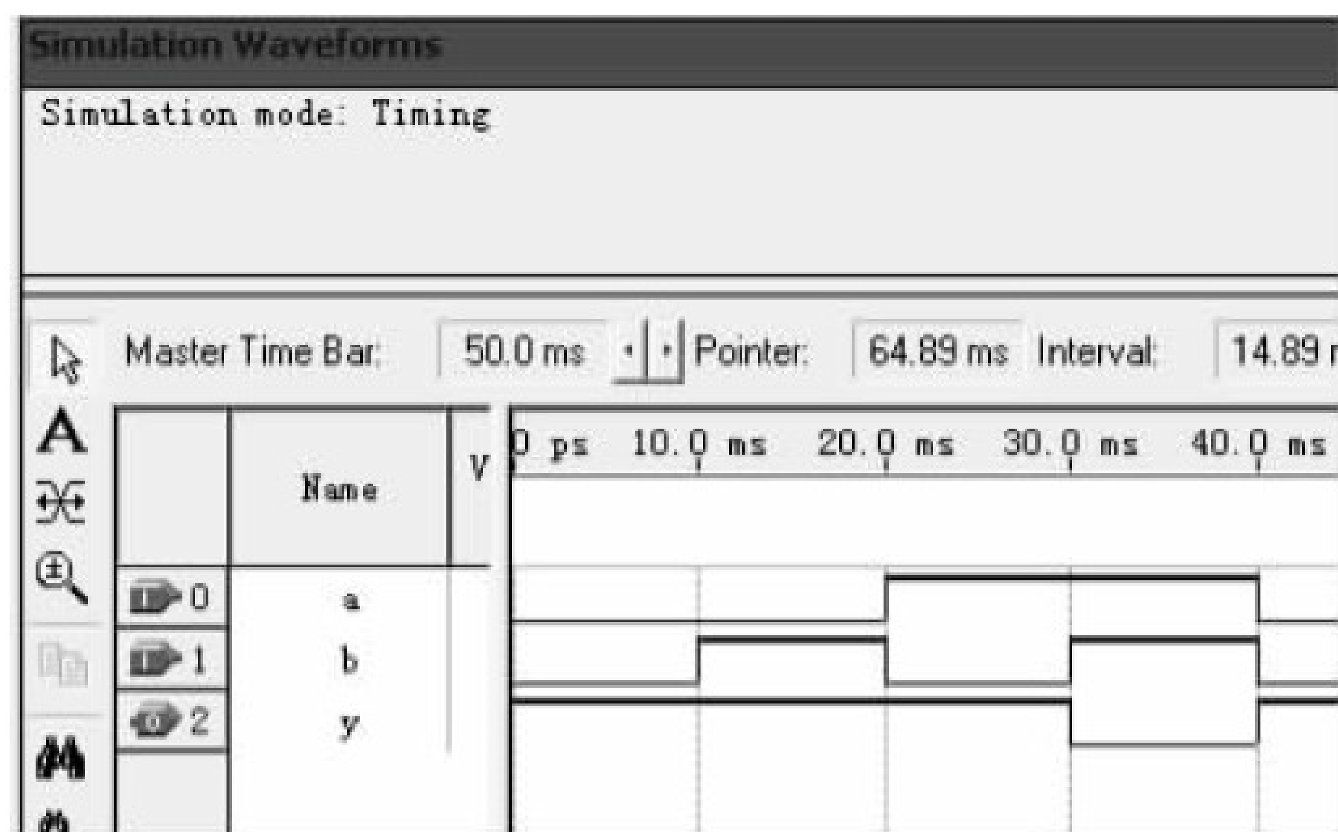


图 3.5.3　例 3.5.3 的仿真波形

习题

3-1　填空题

1. 门具有________、________两种状态，在一定的条件下能够允许信号通过，称为门是被________的。当 TTL 与非门所有输入为 1 时，输出管饱和，处于________状态。

2. 集电极________的门电路，简称为 OC 门。由于 OC 门采用________负载电阻和电源，故可通过选择________的工作电源给负载提供较大的电流，可很好地改善 TTL 门电路的负载能力。

3. CMOS 门电路是由____________MOS 管和____________MOS 管按照________形式连接起来构成的逻辑运算电路(并由此而得名)。CMOS 门电路中，最常用的门电路有________和________。

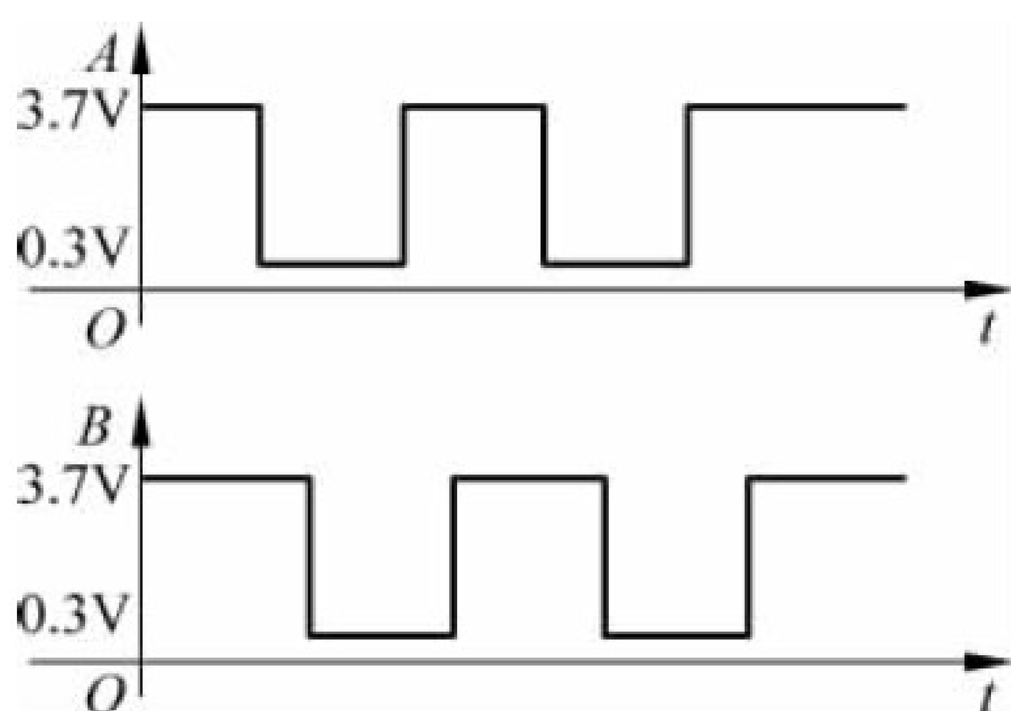
4. 从速度、功耗、工作电压等角度，________/________/________系列是目前 CMOS 系列中性能最后的产品系列，广泛应用于________电子设备，如手机、笔记本、数码相机等。

5. assign 语句是________语句，主要用于对____________变量的赋值。该语句中涉及的________的值一旦发生改变，将立即反映到输出上，这便是 assign 语句中________的含义。

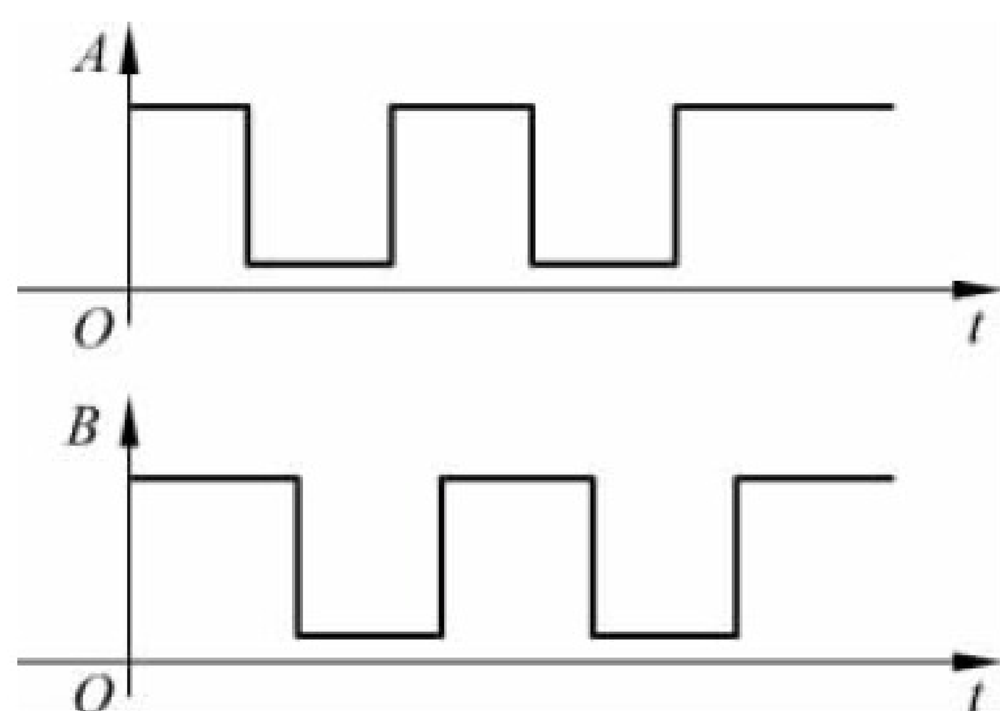
3-2　分析计算题

1. 试画出如图 3.1.1 所示电路的输出端的波形，输入 A、B 的电压波形如题图 3.1 所示。

2. 试画出如题图 3.2 所示输入 A、B 的电压波形加到二输入或非门输入端对应的输出端 Y 的波形。



题图　3.1



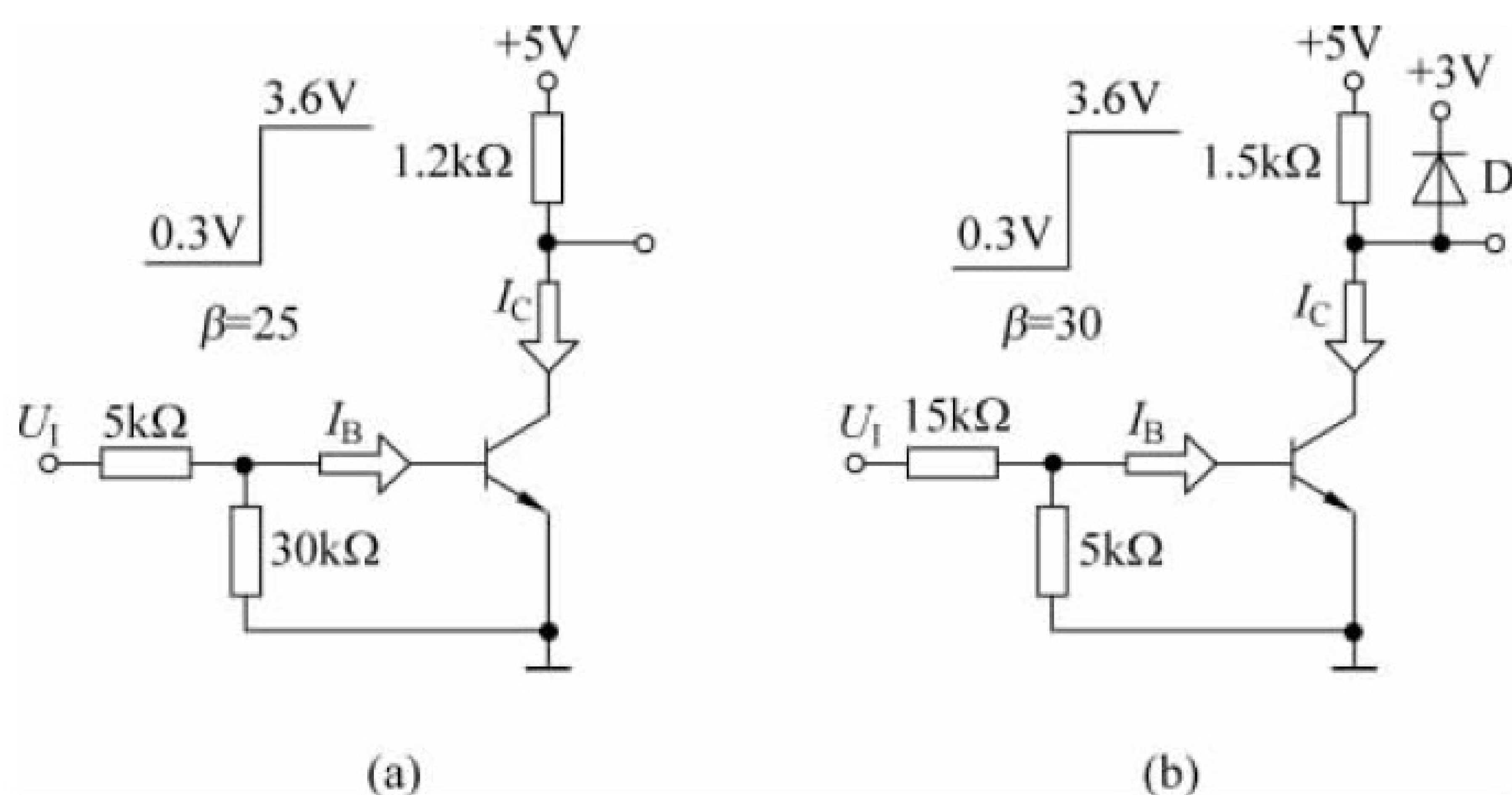
题图　3.2

3. 试分析如题图 3.3 所示电路中三极管的工作状态。

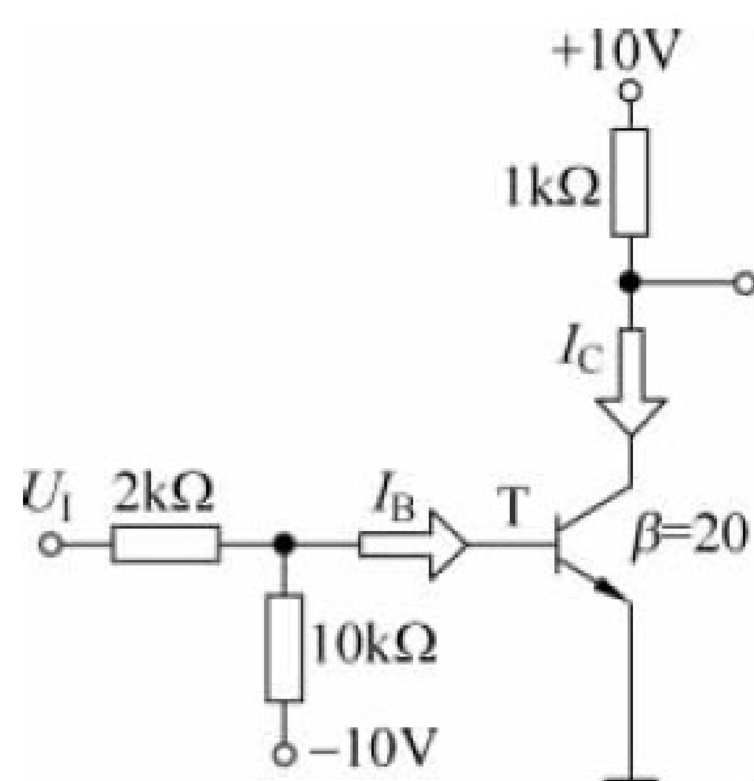
4. 在如题图 3.4 所示的非门电路中：

(1) 若 $U_{BE}=0V$ 时，T 管可靠截止，则允许 U_I 低电平的最大值应为多少？

(2) 要使 T 管临界饱和，则允许 U_I 高电平的最小值应为多少？



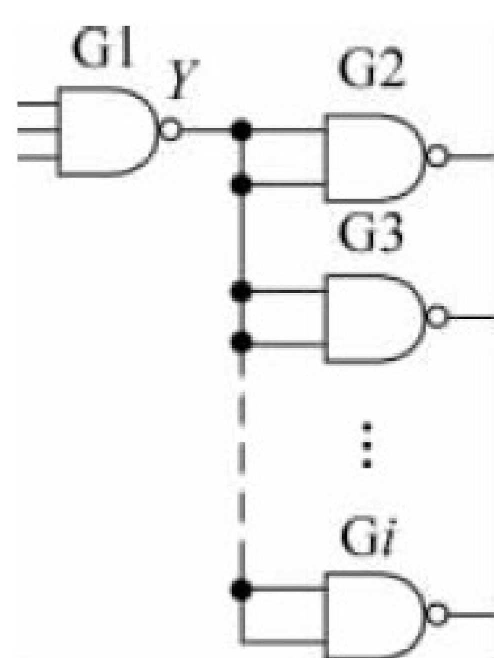
题图 3.3



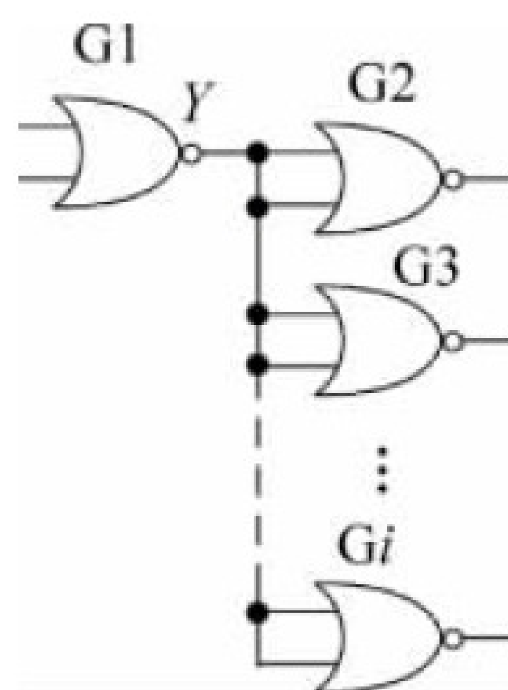
题图 3.4

5. 在如题图 3.5 所示电路中，计算 74 系列 TTL 与非门 G1 最多可驱动多少个同样的与非门电路。其中的与非门电路输入、输出特性分别如图 3.2.4、图 3.2.6、图 3.2.8 所示，要求与非门 G1 输出的高、低电平满足 $U_{OH}\geqslant 3.2V$，$U_{OL}\leqslant 0.3V$。

6. 在如题图 3.6 所示电路中，计算 74 系列 TTL 或非门 G1 最多可驱动多少个同样的或非门电路。其中的或非门电路输入、输出特性分别如图 3.2.4、图 3.2.6、图 3.2.8 所示，要求或非门 G1 输出的高、低电平满足 $U_{OH}\geqslant 3.2V$，$U_{OL}\leqslant 0.3V$。



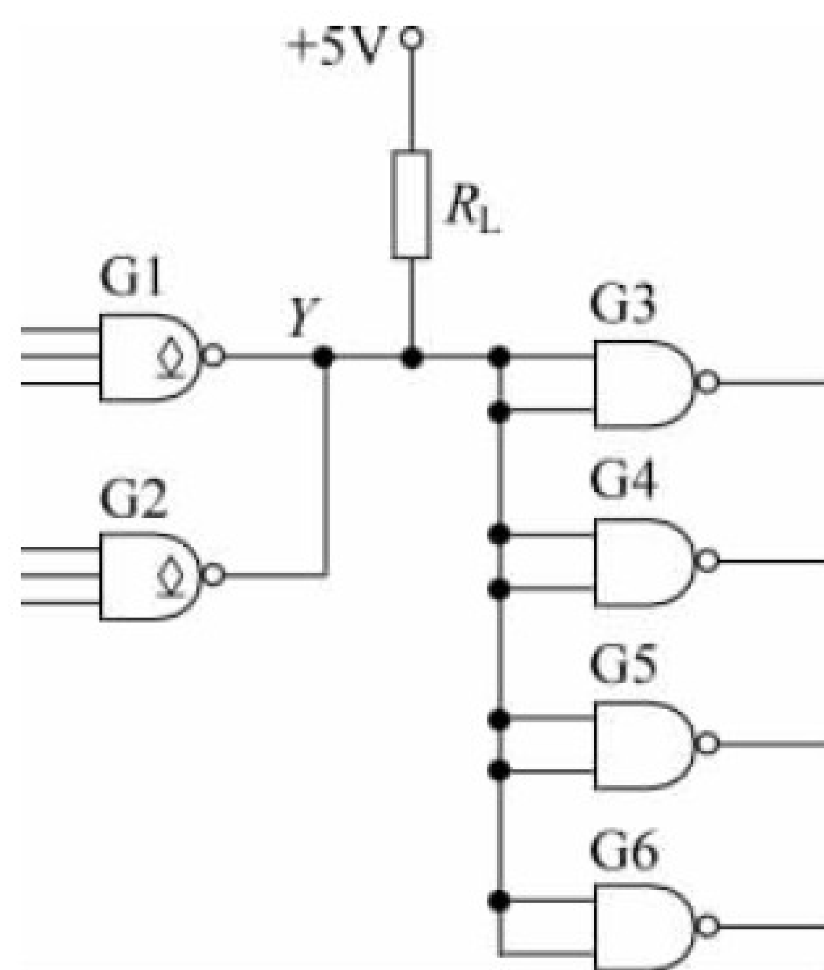
题图 3.5



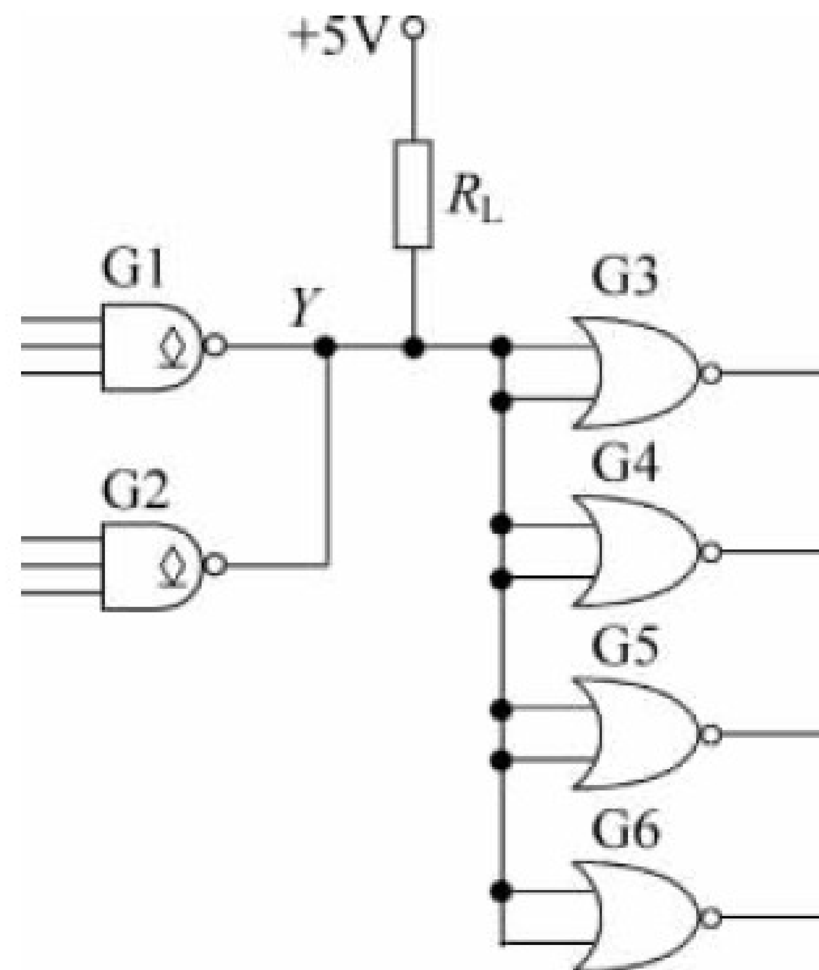
题图 3.6

7. 在如题图 3.7 所示电路中，已知 74 系列 TTL OC 与非门 G1、G2 输出管截止时漏电流 $I_{OH}=0.2mA$，导通时允许的最大负载电流 $I_{OLMAX}=16mA$；与非门 G3、G4、G5、G6 的低电平输入电流 $I_{IL}=-1mA$，高电平输入电流 $I_{IH}=0.04mA$，要求 OC 与非门输出的高、低电平满足 $Y_{OH}\geqslant 3.2V$，$Y_{OL}\leqslant 0.4V$，计算 R_L 的合适值。

8. 在如题图 3.8 所示电路中，已知 74 系列 TTL OC 与非门 G1、G2 输出管截止时漏电流 $I_{OH}=0.1mA$，导通时允许的最大负载电流 $I_{OLMAX}=10mA$；或非门 G3、G4、G5、G6 的低电平输入电流 $I_{IL}=-0.4mA$，高电平输入电流 $I_{IH}=0.02mA$，要求 OC 与非门输出的高、低电平满足 $Y_{OH}\geqslant 3.2V$，$Y_{OL}\leqslant 0.4V$，计算 R_L 的合适值。



题图 3.7



题图 3.8

9. 为什么 TTL 与非门输入端在以下三种接法时，逻辑上都等于输入为 0？

(1) 输入端接地；

(2) 输入端接输出电平为 0.6V 的电源；

(3) 输入端接同类与非门的输出低电平 0.3V。

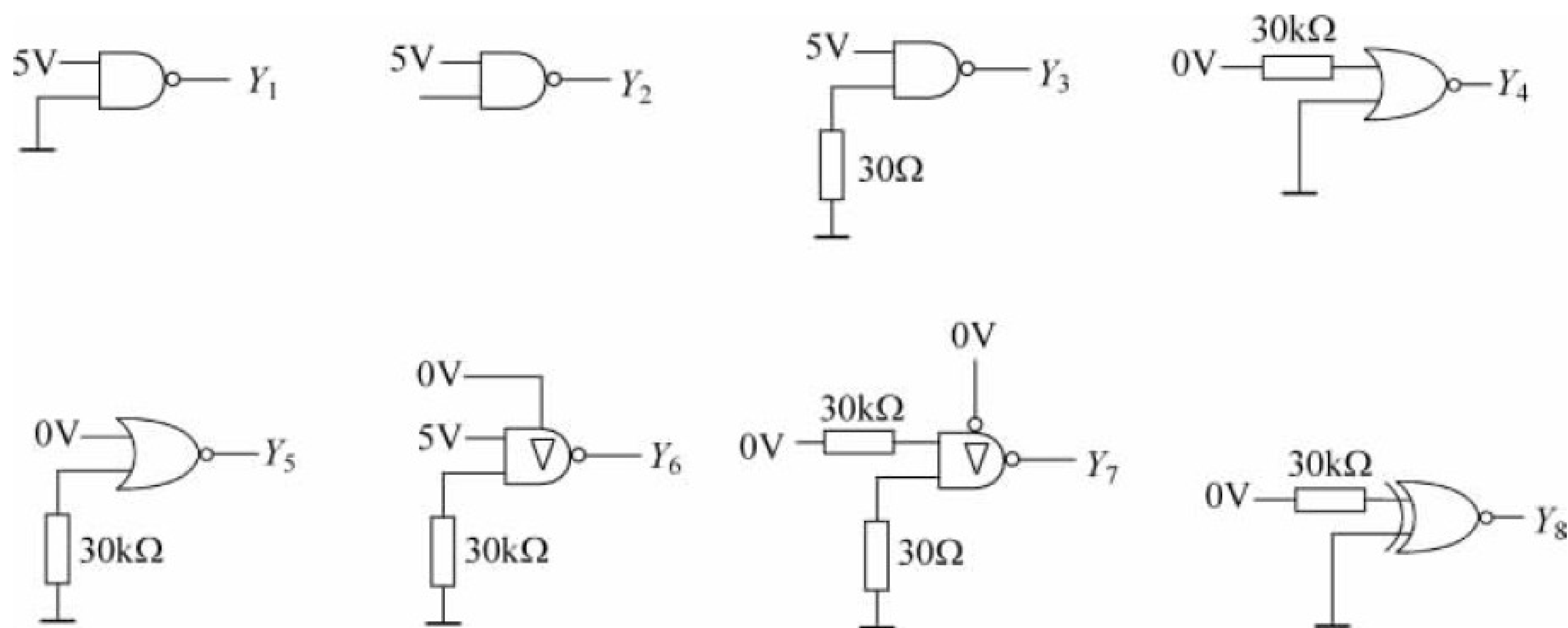
10. 为什么 TTL 与非门输入端在以下三种接法时，在逻辑上都等于输入为 1？

(1) 输入端接同类与非门的输出高电平 3.2V；

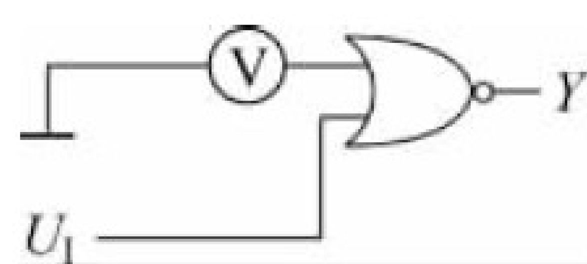
(2) 输入端接高于 2V 的电源；

(3) 输入端悬空。

11. 已知如题图 3.9 所示电路中各门电路均为 TTL 门电路，说出各门电路的输出分别为何状态(高电平、低电平、高阻态)。



题图 3.9



题图 3.10

12. 已知如题图 3.10 所示电路中或非门电路为 TTL 门电路，万用表的内阻为 20kΩ/V，量程为 5V。说出在下列输入情况下万用表的读数。

(1) U_I悬空；

(2) U_I接低电平 0.2V；

(3) U_I接高电平 3.2V；

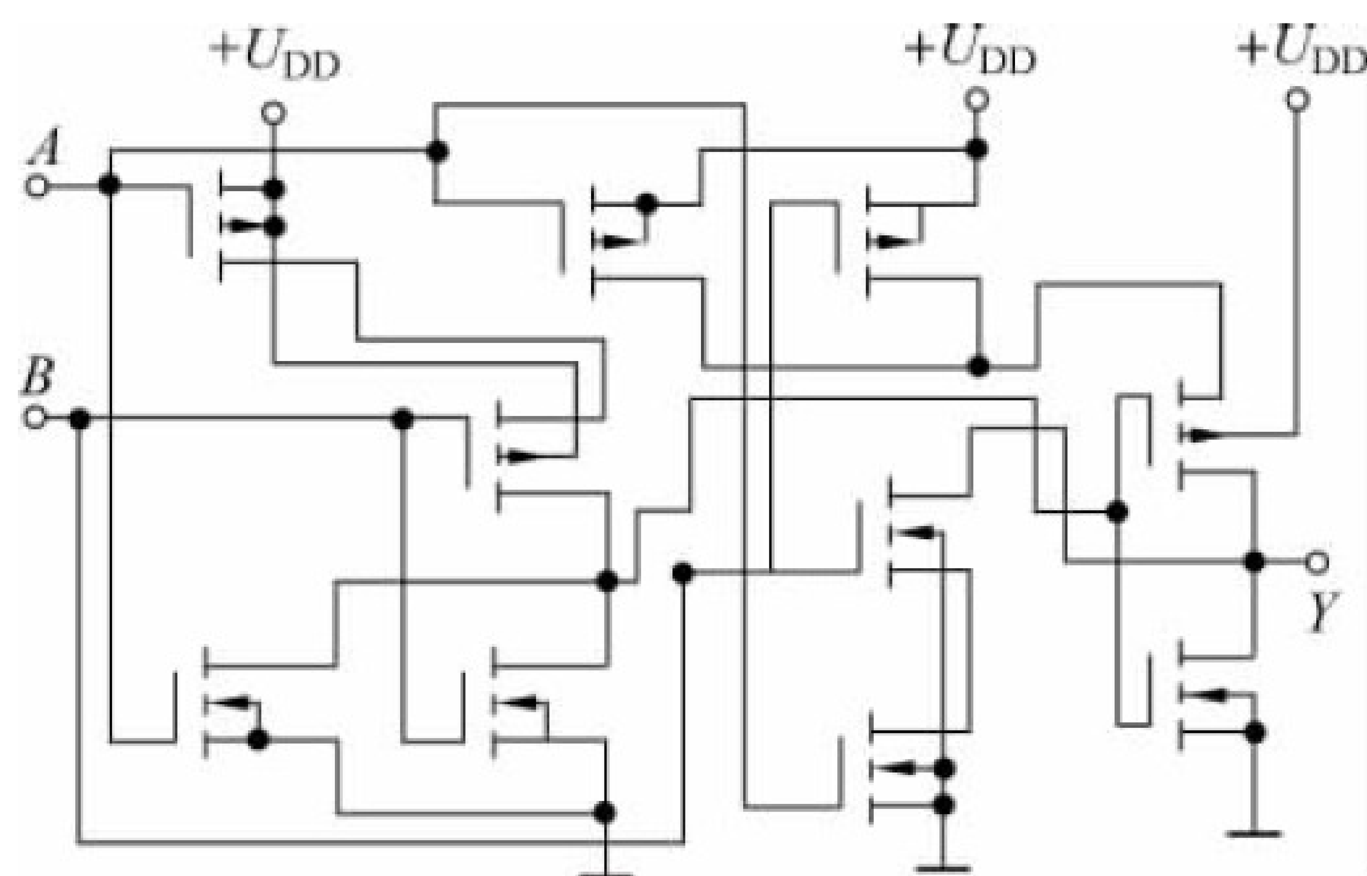
(4) U_I经 51Ω 电阻接地；

(5) U_I经 50kΩ 电阻接地。

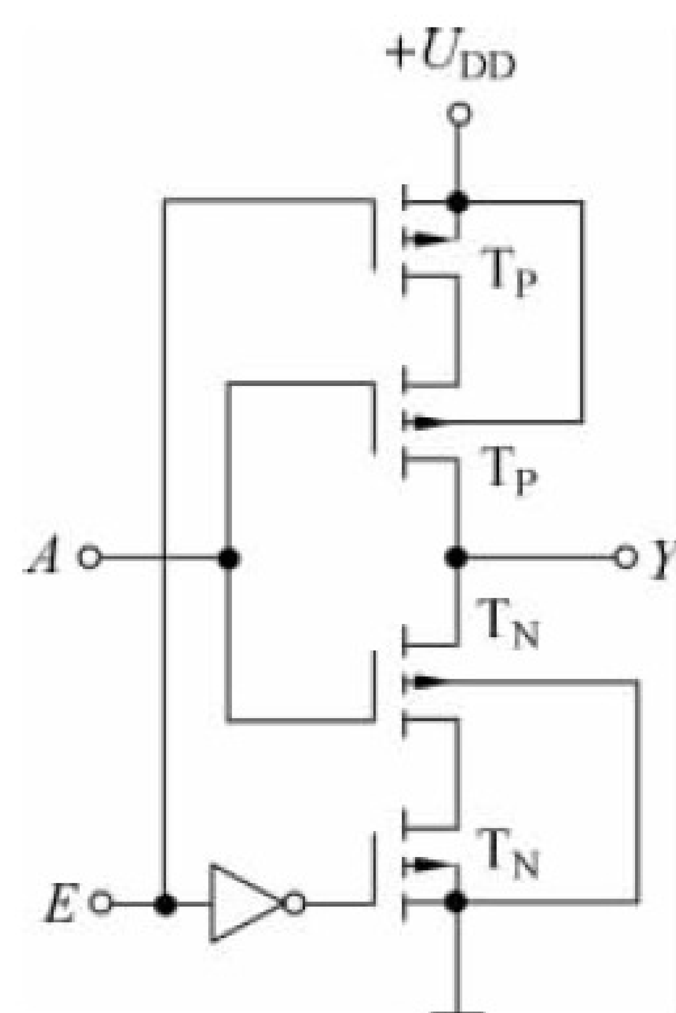
13. 分析计算题 12 中，将或非门改为与非门。说出在上述 5 种输入情况下万用表的读数。

14. 分析题图 3.11 所示电路的逻辑功能。

15. 分析题图 3.12 所示电路的逻辑功能。



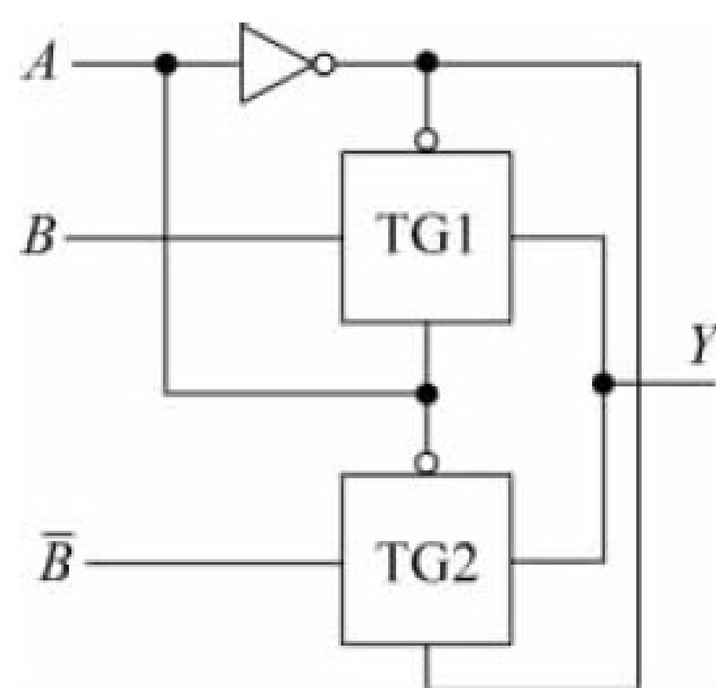
题图　3.11



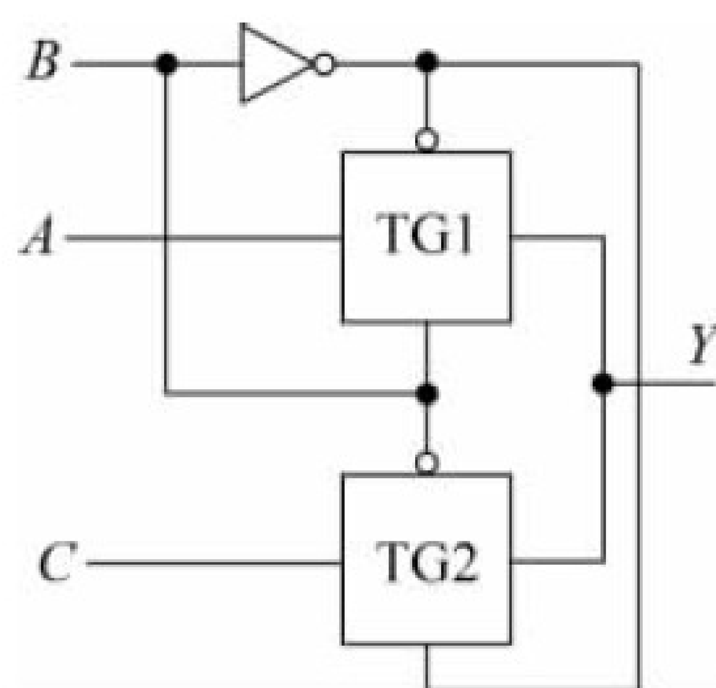
题图　3.12

16. 分析题图 3.13 所示电路的逻辑功能。

17. 分析题图 3.14 所示电路的逻辑功能。



题图　3.13



题图　3.14

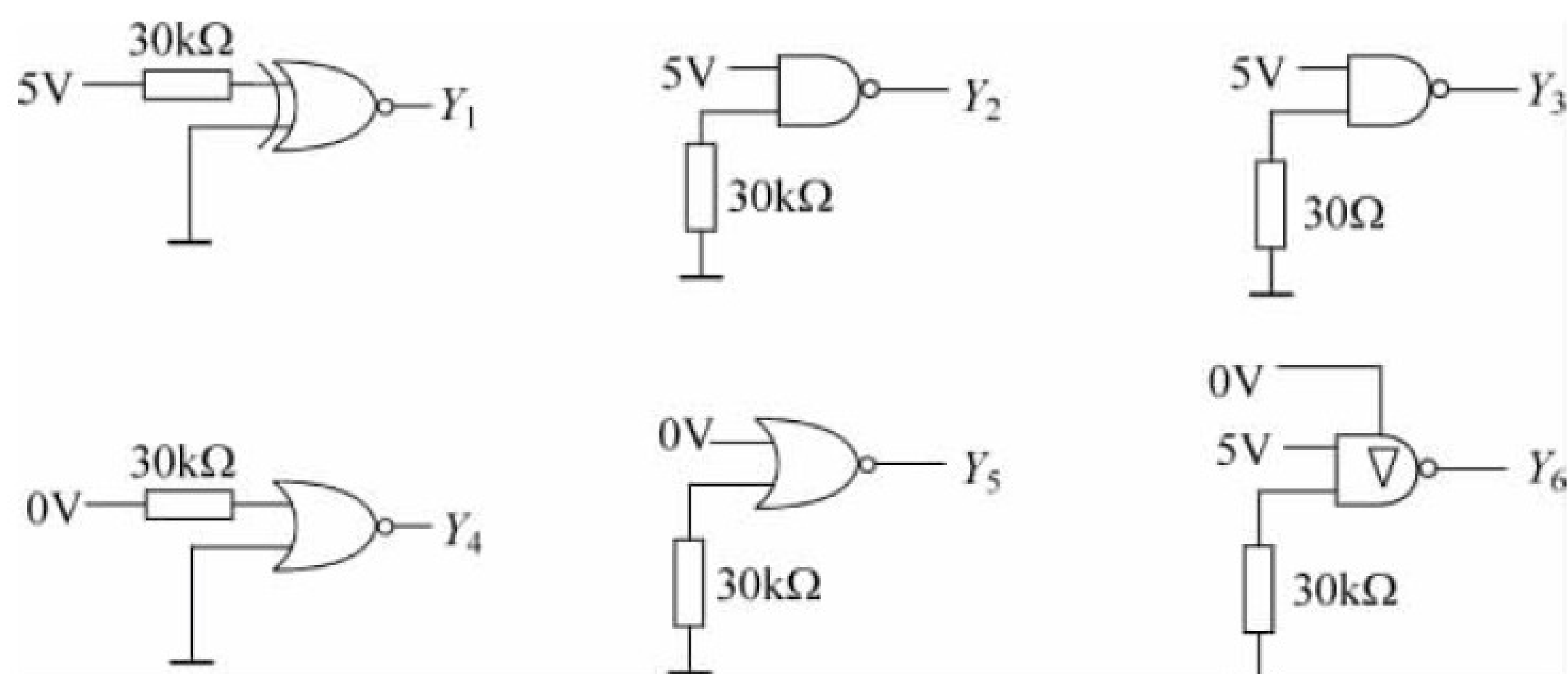
18. 参考例 3.3.3 实现一个二输入与非门。

19. 参考例 3.3.3 实现一个二输入或非门。

20. 分析计算题 13 中的与非门为 CMOS 与非门，电源 U_{DD}＝5V，说出在该题中 5 种输入情况下万用表的读数。

21. 已知如题图 3.15 所示电路中各门电路均为 CMOS 门电路，电源 U_{DD}＝5V，说出各门电路的输出分别为什么状态。

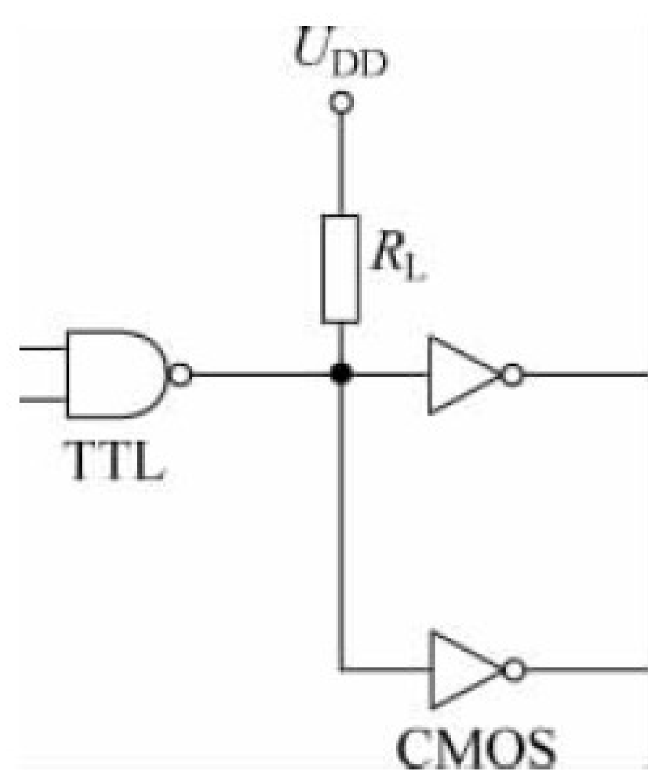
22. 如题图 3.16 所示为用 TTL 门电路驱动 CMOS 门电路的实例，试计算上拉电阻 R_L 的取值范围。已知 TTL 与非门输出为低电平(0.3V)时，最大输出电流为 8mA，输出为高



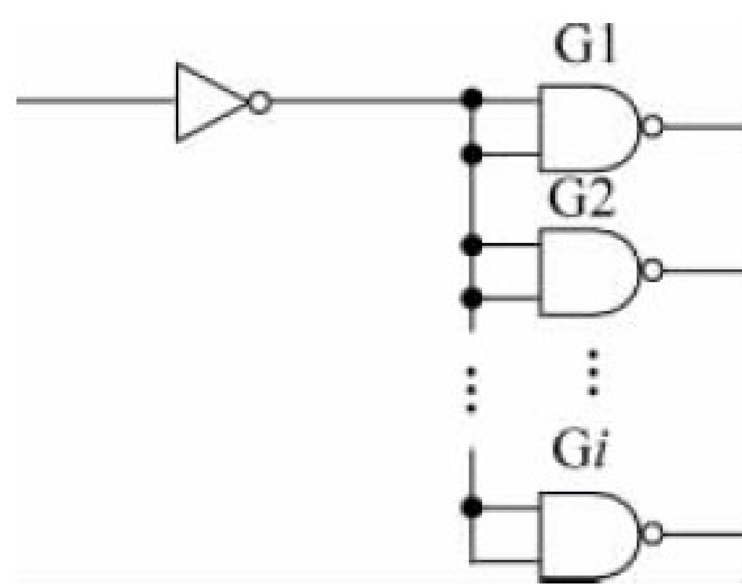
题图 3.15

电平时漏电流为 50μA。CMOS 反相器的高电平输入电流和低电平输入电流的最大值均为 1μA。要求加到 CMOS 反相器输入端的电压满足高电平不小于 4V，低电平不大于 0.3V，电源工作电压 5V。

23. 如题图 3.17 所示为用 CMOS 门电路 74HCT04 驱动 TTL 门电路的实例，试计算当 TTL 与非门分别为 7400、74LS00、74ALS00 时，CMOS 门电路所能驱动的 TTL 与非门的个数。74HCT04 性能参数见表 3.4.1，7400、74LS00、74ALS00 性能参数见表 3.2.3。



题图 3.16



题图 3.17

24. 试说出下列各门电路中哪些可以将输出端并联使用。

(1) 标准 74 系列互补输出门电路；

(2) TTL 电路的 OC 门；

(3) TTL 三态输出门电路；

(4) 标准互补输出 CMOS 门电路；

(5) CMOS 电路的 OD 门；

(6) CMOS 三态输出门电路。

25. 画出下面代码描述的逻辑门对应的逻辑符号。

```
module ort3(a,b,c,y);
  input   a,b,c;
  output   y;
  assign y = a|b|c;
endmodule
```

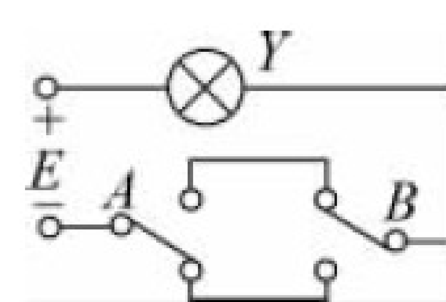
26. 画出下面代码描述的逻辑门对应的逻辑符号。

```
module ant3(a,b,c,y);
  input  a,b,c;
  output  y;
  assign y = a & b & c;
endmodule
```

27. 用 Verilog 代码描述一个三输入或非门。

3-3　应用题

1. 如题图 3.18 所示为一控制楼梯照明的有触点电路，在楼上、楼下各装一个单刀双掷开关 A 和 B，这样人在楼上和楼下都可以开灯和关灯。设 $Y=1$ 表示灯亮，$Y=0$ 表示灯灭；$A=1$ 表示开关向上扳，$A=0$ 表示开关向下扳，B 亦如此。试用 Verilog 代码描述该逻辑问题。



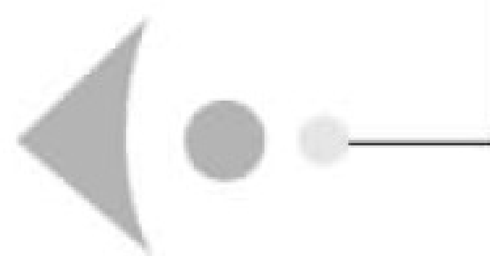
题图　3.18

2. 试用 CMOS 反相器和传输门实现应用题 1 中描述的逻辑问题。



第4章

组合逻辑电路





本章要点：

本章为本书重点章。学习本章应重点理解组合逻辑电路的特征及其分析方法，理解编码器、译码器、数据选择器、数值比较器、加法器等常用组合逻辑电路的逻辑功能特点及其相应集成芯片的使用方法；掌握用译码器、数据选择器实现组合逻辑电路的方法，初步理解组合逻辑电路的计算机仿真分析方法，懂得竞争-冒险的含义及其简单的检查与消除方法。

4.1 概述



数字电路根据逻辑功能的不同，可以分为两大类：一类为组合逻辑电路(Combinational Logic Circuit，简称组合电路)；另一类为时序逻辑电路(Sequential Logic Circuit，简称时序电路)。时序电路将在第6章中进行讨论。

所谓组合逻辑电路，是指任意时刻电路输出的逻辑值，仅取决于该瞬间电路输入的逻辑值，而与电路的原状态无关。例如，竞赛抢答器、常用的编码器、译码器、全加器、数值比较器、数据选择器、奇偶产生器/奇偶校验器等都属于组合逻辑电路。

可从电路所具有的逻辑功能角度理解为什么该电路为组合逻辑电路。如4路输入竞赛抢答器。4路输入分别对应1、2、3、4四个选手。当前抢答状态为1号灯亮，1号选手成功获得抢答机会。为什么是这样呢？因为1号选手最先按下按钮，没有其他原因，即输出只与该时刻的输入有关。

当然，电路的逻辑功能主要是通过门电路之间的具体连接关系来实现的，因此，也可通过门电路之间的具体连接方法来判断该电路是组合逻辑电路还是时序逻辑电路。

根据组合逻辑电路的含义，其输出仅仅取决于该时刻的输入信号，而与以前的输出(电路的原状态)无关。从连接角度，电路的输出与以前的输出有关的原因是以前的输出反馈

到了电路的输入端。可见，组合电路的连接特征为“只有输入到输出的通路，没有输出到输入的反馈”。

根据上面的分析，如图 4.1.1 所示电路中，图(a)为组合逻辑电路，图(b)为时序逻辑电路。

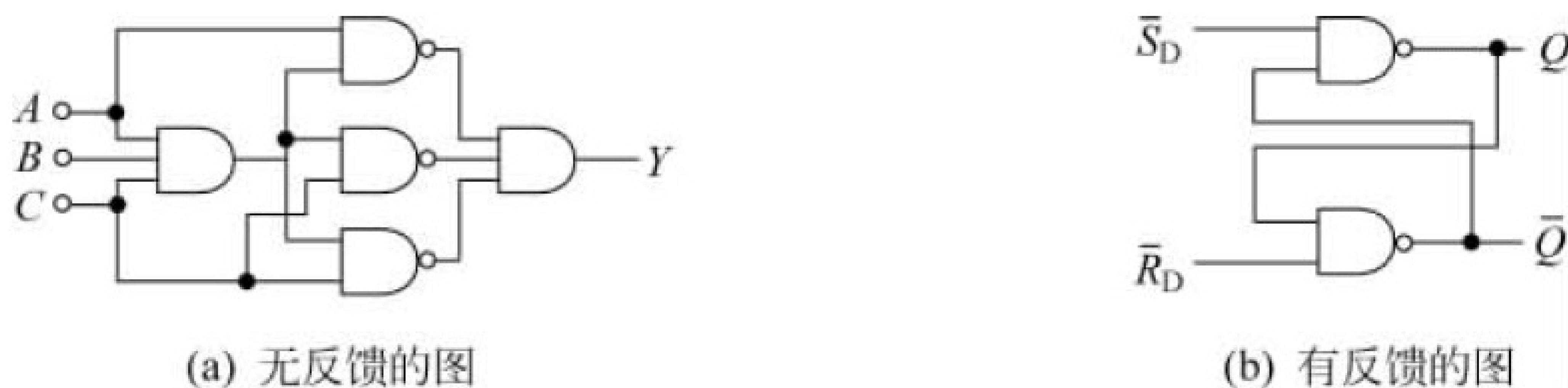


图 4.1.1　判断组合逻辑电路

根据组合电路的连接特征，对具有 i 个输入、j 个输出的组合电路，可用如图 4.1.2 所示的框图来描述。



图 4.1.2　组合逻辑电路的框图

可见，对于 i 个输入、j 个输出的组合电路，可以用 j 个输出方程来描述该电路的逻辑功能。即

$$\begin{cases} Y_1 = F_1(X_1,\cdots,X_i) \\ \vdots \\ Y_j = F_j(X_1,\cdots,X_i) \end{cases} \tag{4.1.1}$$

当然，组合电路除用表达式描述外，还常用真值表、逻辑图、工作波形图等来描述。

复习思考题

同学小王用手机通过通信录联系人“小李”呼叫拨打小李手机，请问这个呼叫应用为组合电路应用还是时序电路应用？为什么？

4.2　组合逻辑电路的分析和设计

根据已知的组合逻辑电路，寻找出该电路所实现的逻辑功能，称为组合逻辑电路的分析。根据给出的实际组合逻辑问题，完成实现这一逻辑功能的最简逻辑电路，称为组合逻辑电路的设计。

4.2.1　组合逻辑电路的分析



组合逻辑电路具有多种表示形式。已知组合逻辑问题的逻辑电路图，可通过该电路图直接求出该电路对应的逻辑表达式。许多场合下，逻辑表达式不能直观反映电路的逻辑功能，应根据已知的电路，写出相应的真值表。

可总结组合逻辑电路的分析步骤如下：

(1) 由已知的电路写出电路输出的逻辑函数表达式。具体求解表达式时，可从输出开始逐级写出各门的表达式，也可从输入开始逐级写出各门的输出逻辑式，进而求出最终表

达式。为求解简洁，可用公式法或卡诺图将逻辑函数化简。

(2) 由(1)给出的逻辑表达式填出输出函数的真值表。

(3) 根据真值表叙述该电路所实现的逻辑功能。

(4) 利用仿真软件求出该电路的仿真波形并进一步分析该电路的逻辑功能。

下面通过几个例题介绍组合逻辑电路的分析方法。

【例 4.2.1】 试分析如图 4.2.1 所示电路的逻辑功能。

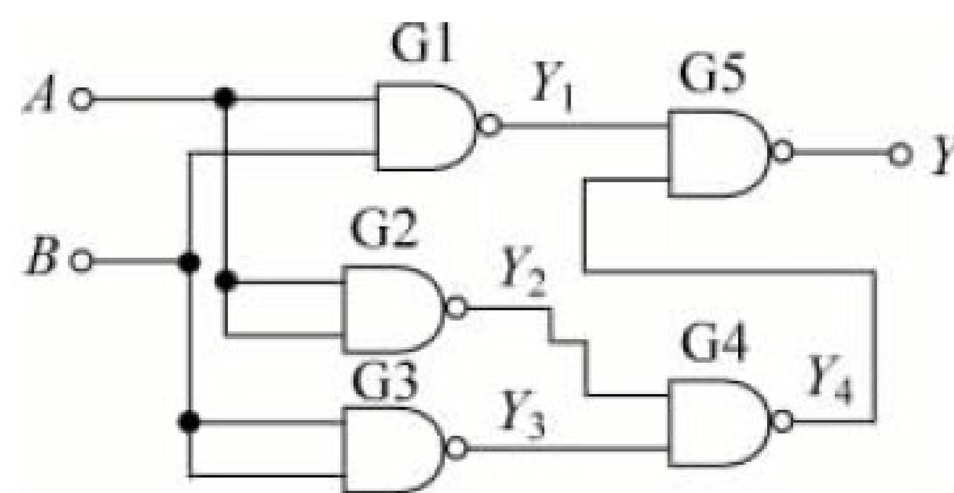


图 4.2.1 例 4.2.1 的图 1

解 (1) 根据电路写出电路输出的逻辑表达式，方法：由电路的输入端到输出端，逐步写出各个门的输出逻辑式，最后写出电路输出 Y 的逻辑表达式。

具体如下：

$$\text{G1}: Y_1 = \overline{AB}$$

$$\text{G2}: Y_2 = \overline{A}$$

$$\text{G3}: Y_3 = \overline{B}$$

$$\text{G4}: Y_4 = \overline{\overline{A}\,\overline{B}}$$

$$\text{G5}: Y = \overline{Y_1 Y_4} = \overline{\overline{AB}\;\overline{\overline{A}\,\overline{B}}}$$



(2) 表达式不简洁，利用摩根定理变换为与或式，有

$$Y = \overline{\overline{AB}\;\overline{\overline{A}\,\overline{B}}} = \overline{\overline{AB}} + \overline{A}\,\overline{B} = AB + \overline{A}\,\overline{B}$$

(3) 根据化简后的表达式，填出函数 Y 的真值表，如表 4.2.1 所示。

表 4.2.1 例 4.2.1 的真值表

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

(4) 结论。由真值表可看出，当输入端 A 和 B 同时为 1 或同时为 0 时，电路输出为 1；否则为 0。这种电路称为“同或门”电路。其逻辑表达式也可写成

$$Y = A \odot B$$

(5) 计算机仿真分析。将本例电路输入到 QuartusⅡ中编译并仿真，得到的波形如图 4.2.2[①]所示。

由仿真波形可看出，当 A、B 相同时输出为 1；否则为 0，具有“同或门”的逻辑功能。

---

① QuartusⅡ等 EDA 软件不支持斜体、下标等文本格式，不符合出版规范或本书的书写习惯。但为使读者更易于理解仿真结果，本章中的仿真结果图采用原始仿真图，特此说明。

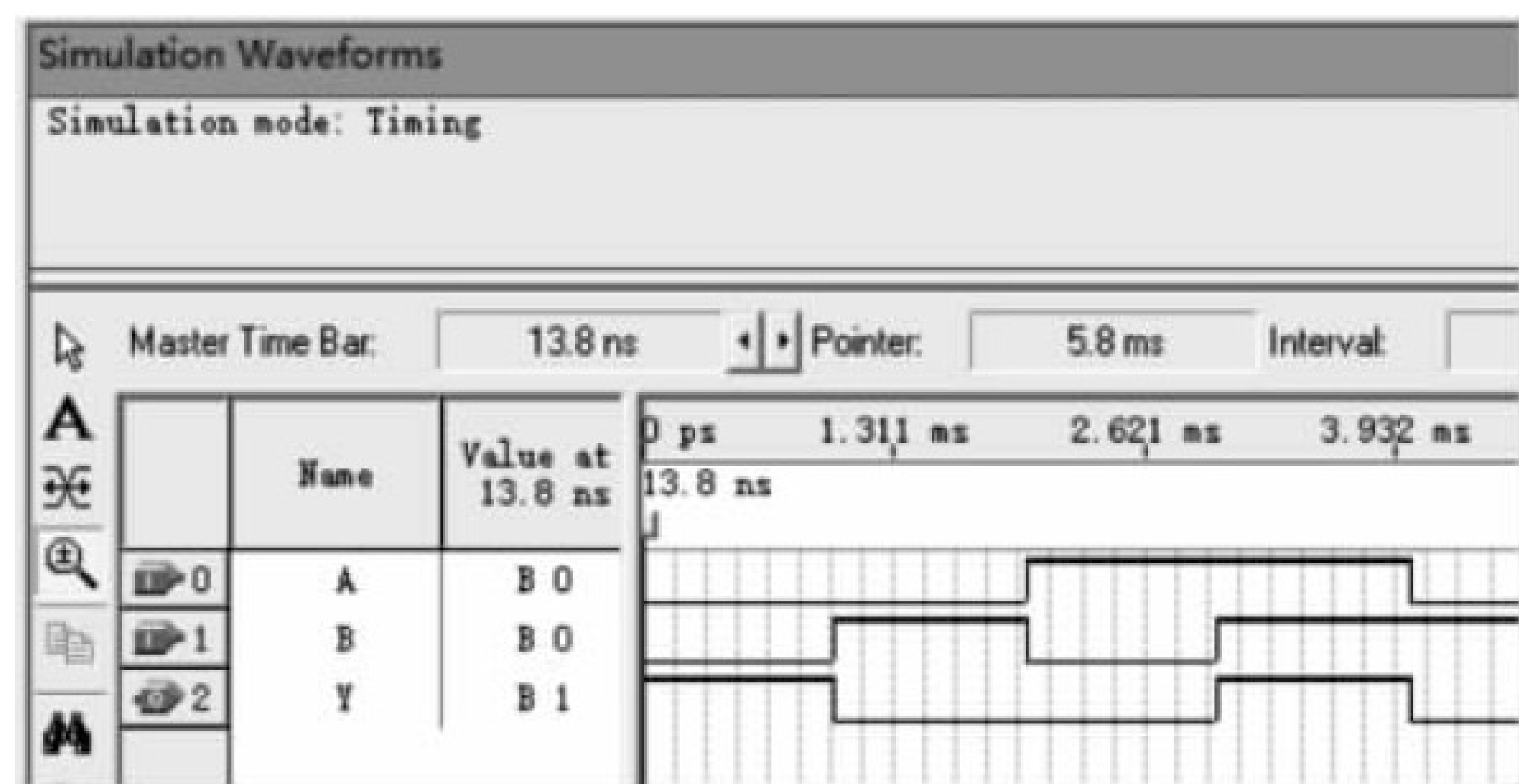


图 4.2.2　例 4.2.1 的图 2

【例 4.2.2】　试分析如图 4.2.3 所示电路的逻辑功能。

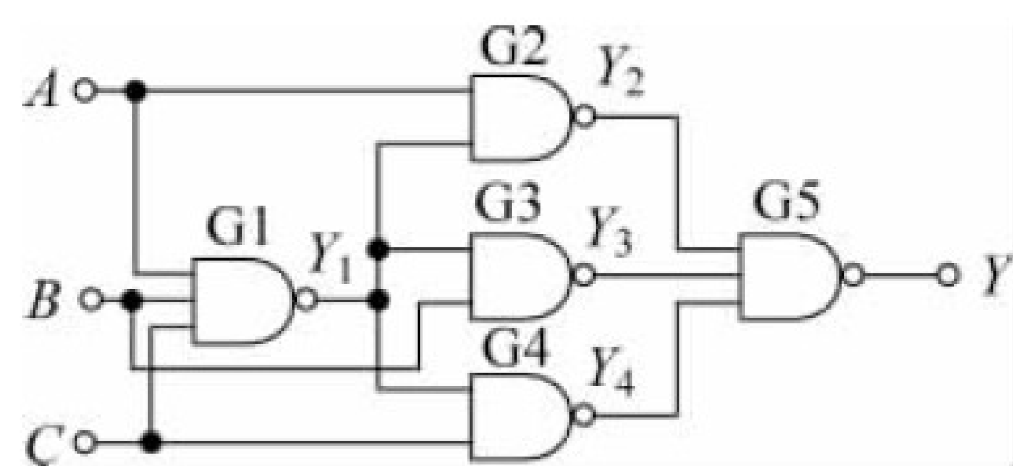


图 4.2.3　例 4.2.2 的图 1

解　(1) 根据电路写出电路输出的逻辑表达式。由电路的输出端到输入端，逐步写出各个门的输出逻辑式，可求出最终表达式如下：

$$\text{G5}: Y=\overline{Y_2Y_3Y_4}=\overline{\overline{A\,\overline{ABC}}\cdot\overline{B\,\overline{ABC}}\cdot\overline{C\,\overline{ABC}}}$$

(2) 表达式不简洁，利用摩根定理变换为与或式，有

$$\begin{aligned}Y&=\overline{\overline{A\,\overline{ABC}}\cdot\overline{B\,\overline{ABC}}\cdot\overline{C\,\overline{ABC}}}=A\,\overline{ABC}+B\,\overline{ABC}+C\,\overline{ABC}\\&=A\,\overline{BC}+B\,\overline{AC}+C\,\overline{AB}=A(\bar{B}+\bar{C})+B(\bar{A}+\bar{C})+C(\bar{A}+\bar{B})\\&=A\bar{B}+A\bar{C}+\bar{A}B+B\bar{C}+\bar{A}C+\bar{B}C\\&=A\bar{B}+\bar{A}C+B\bar{C}\end{aligned}$$

(3) 根据化简后的表达式，填出函数 Y 的真值表，如表 4.2.2 所示。

表 4.2.2　例 4.4.2 的真值表

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

(4) 结论。由真值表可看出，当电路输入端 A、B、C 不完全相同时，电路输出 Y 为 1；否则，输出 Y 为 0。

该电路又称为三变量不一致判别电路。

(5) 计算机仿真。将本例电路输入到 QuartusⅡ中编译并仿真，得到的波形如图 4.2.4 所示。

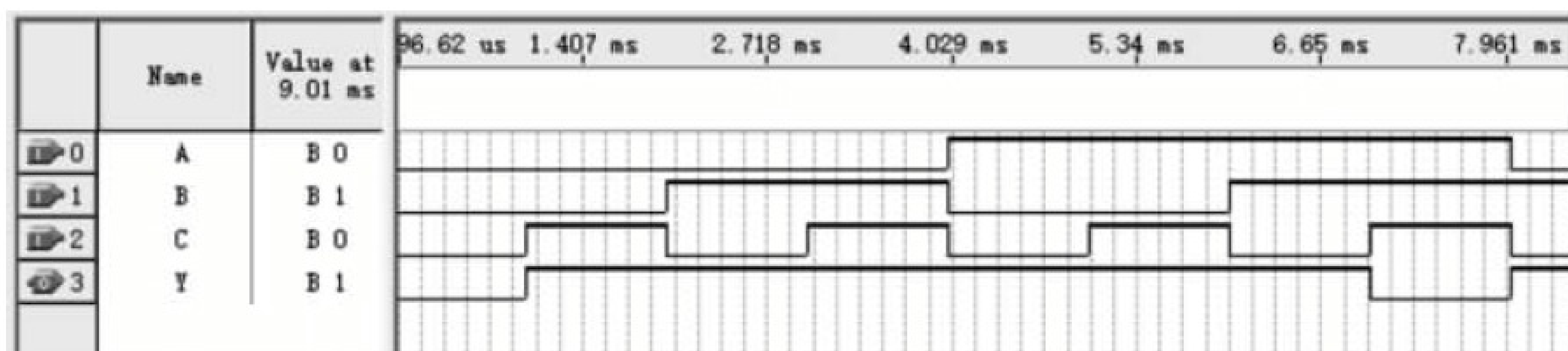


图 4.2.4　例 4.2.2 的图 2

由仿真波形可看出，该电路具有“三变量不一致判别”的逻辑功能。

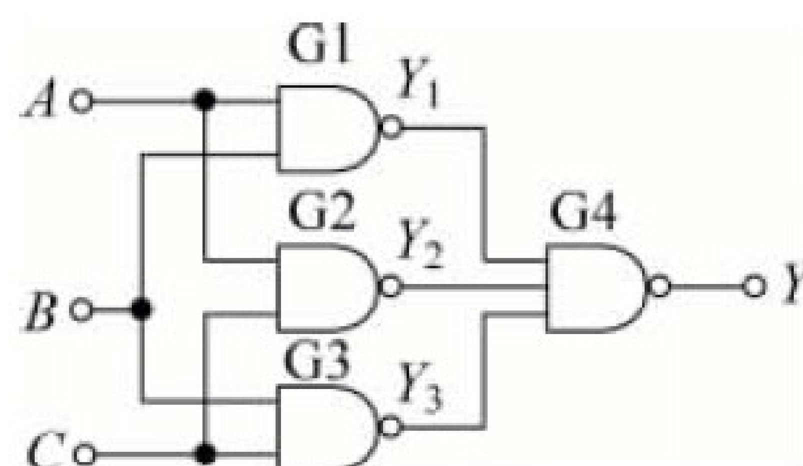


图 4.2.5　例 4.2.3 的图 1

【例 4.2.3】 试分析如图 4.2.5 所示电路的逻辑功能。

解　(1) 根据电路写出电路输出的逻辑表达式。具体如下：

$$\text{G4}:\ Y=\overline{Y_2Y_3Y_1}=\overline{\overline{AC}\ \overline{BC}\ \overline{AB}}$$

(2) 表达式不简洁，利用摩根定理变换为与或式，有

$$Y=\overline{Y_2Y_3Y_1}=\overline{\overline{AC}\ \overline{BC}\ \overline{AB}}$$
$$=AB+AC+BC$$

(3) 根据简化后的表达式，填出函数 Y 的真值表，如表 4.2.3 所示。

表 4.2.3　例 4.2.3 的真值表

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(4) 结论。由真值表可看出，这是一个 3 人表决逻辑电路。即 3 个人对一个提案进行表决，赞成该提案用 1 表示，不赞成该提案用 0 表示。如果有 2 个或者 2 个以上的人赞成该提案，则该提案就被通过，电路 Y 输出 1；否则，该提案被否决，电路输出 Y 为 0。此电路又称为多数表决电路。

在实际应用中，每一个人都有一个按钮，如果对提案赞成就按按钮，表示 1，如果不赞成该提案，就不按按钮，表示 0。其表决结果用指示灯来表示，如果多数赞成，则指示灯亮，$Y=1$；反之则不亮，$Y=0$。

(5) 计算机仿真。将本例电路输入到 QuartusⅡ中编译并仿真，得到的波形如图 4.2.6 所示。由仿真波形可看出，该电路具有“3 人表决结果输出”的逻辑功能。

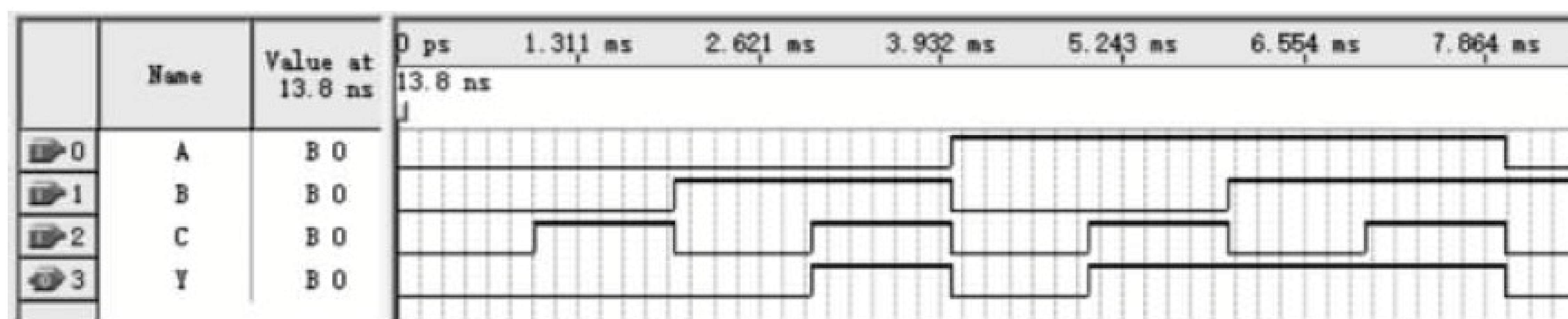


图 4.2.6　例 4.2.3 的图 2

*4.2.2　利用 Quartus Ⅱ 分析组合逻辑电路



下面以如图 4.2.1 所示电路为例，利用仿真软件 Quartus Ⅱ 9.0 SP2 介绍 Quartus Ⅱ 分析组合逻辑电路的方法。

1. 建立仿真项目的工程文件

启动 Quartus Ⅱ，选择 File→New 子菜单，在随后弹出的任务窗口中选择文件类型为 Block Diagram/Schematic File，单击 OK 按钮进入逻辑图形文件编辑状态。

选择 File→Save As 子菜单，在随后弹出的“另存文件任务”窗口中为新创建的未命名的图形文件取一个适当的名字(如 E4_2_1。因 Quartus Ⅱ 仿真时要产生文件，因此最好为仿真项目新建一个子目录)，单击“保存”按钮。保存前应确认已勾选窗口最下方的 Create new project based on this file 复选框。在随后弹出的“创建工程文件确认对话框”中单击“是”按钮进入“工程创建任务”窗口，如图 4.2.7 所示，可以在该窗口中选择工作目录，也可单击 Finish 按钮完成工程文件的创建。

2. 建立要仿真的逻辑图形文件并编译

如图 4.2.1 所示电路包括 5 个与非门、2 个输入、1 个输出。具体实现如下。

(1) 在编辑区任意位置双击，将弹出“逻辑符号放置任务”窗口，单击任务窗口左边的“＋”，展开 primitives→logic 库，如图 4.2.8 所示。

移动滚动条，选择 and2 元件(二输入与非门)，单击 OK 按钮，与非门符号便出现在绘图区，单击具体位置确认放置 1 个与非门。

(2) 可依照上述方法放置 5 个与非门。也可单击选择与非门，单击右键，在弹出的菜单中选择 Copy，复制与非门。在需要粘贴与非门的适当位置右击鼠标，在弹出的菜单中选择 Paste，复制 4 个与非门。拖动与非门到合适位置，如图 4.2.9 所示。

(3) 放置输入、输出符号。在编辑区任意位置双击，按(1)中的方法展开 pin 库，选择 input 元件，单击 OK 按钮，输入符号便出现在绘图区，单击具体位置确认放置 1 个输入，继续放置另一个输入。选择 output 元件，放置另一个输出，如图 4.2.10 所示。

(4) 定义元件名称。系统放置元件时，自动定义了 1 个默认名称，可将元件定义为要求的名称。双击输入元件中的文字 pin_name，将弹出“引脚属性任务”窗口，在 Pin name(s)文

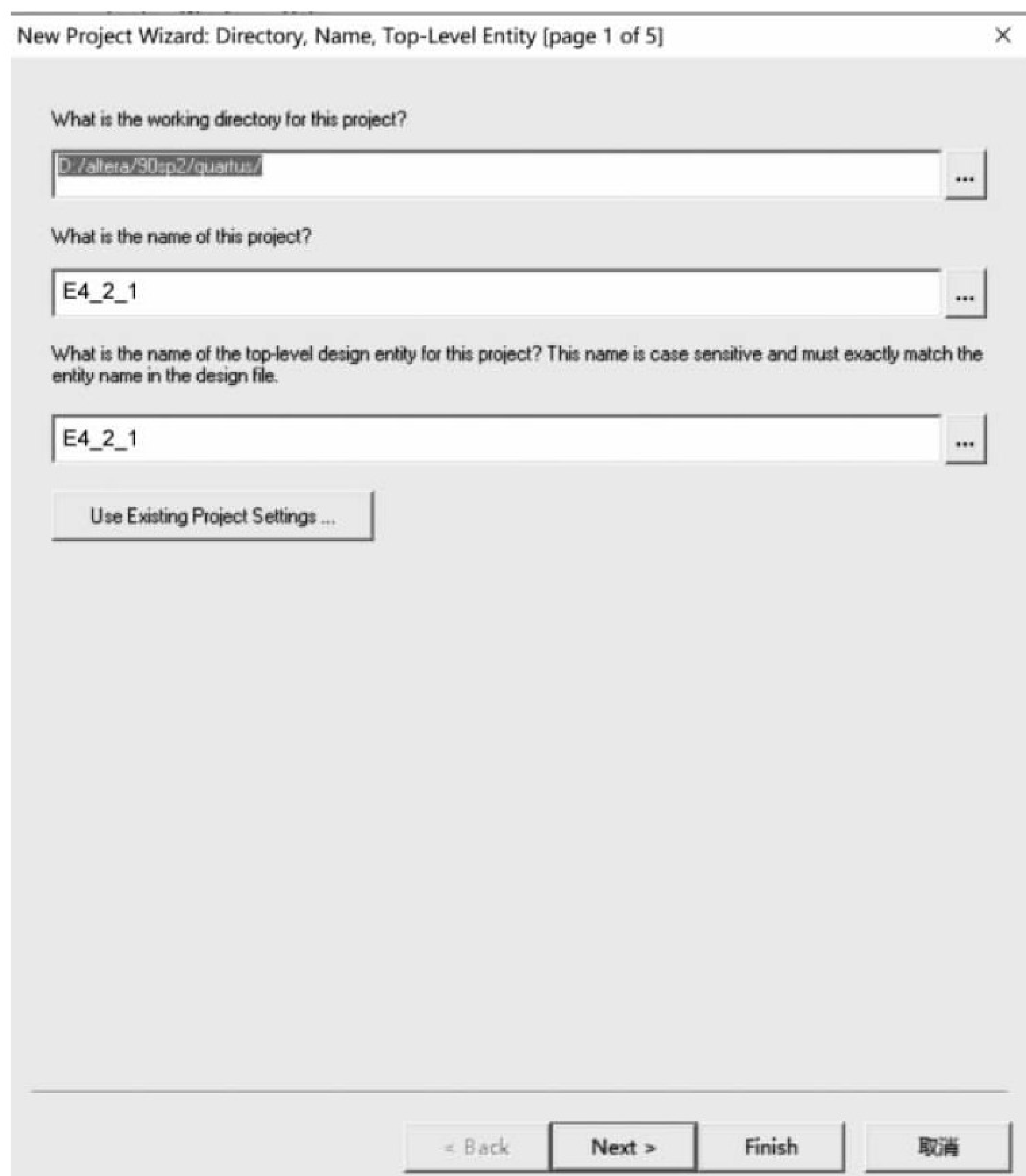


图 4.2.7　工程创建参考界面

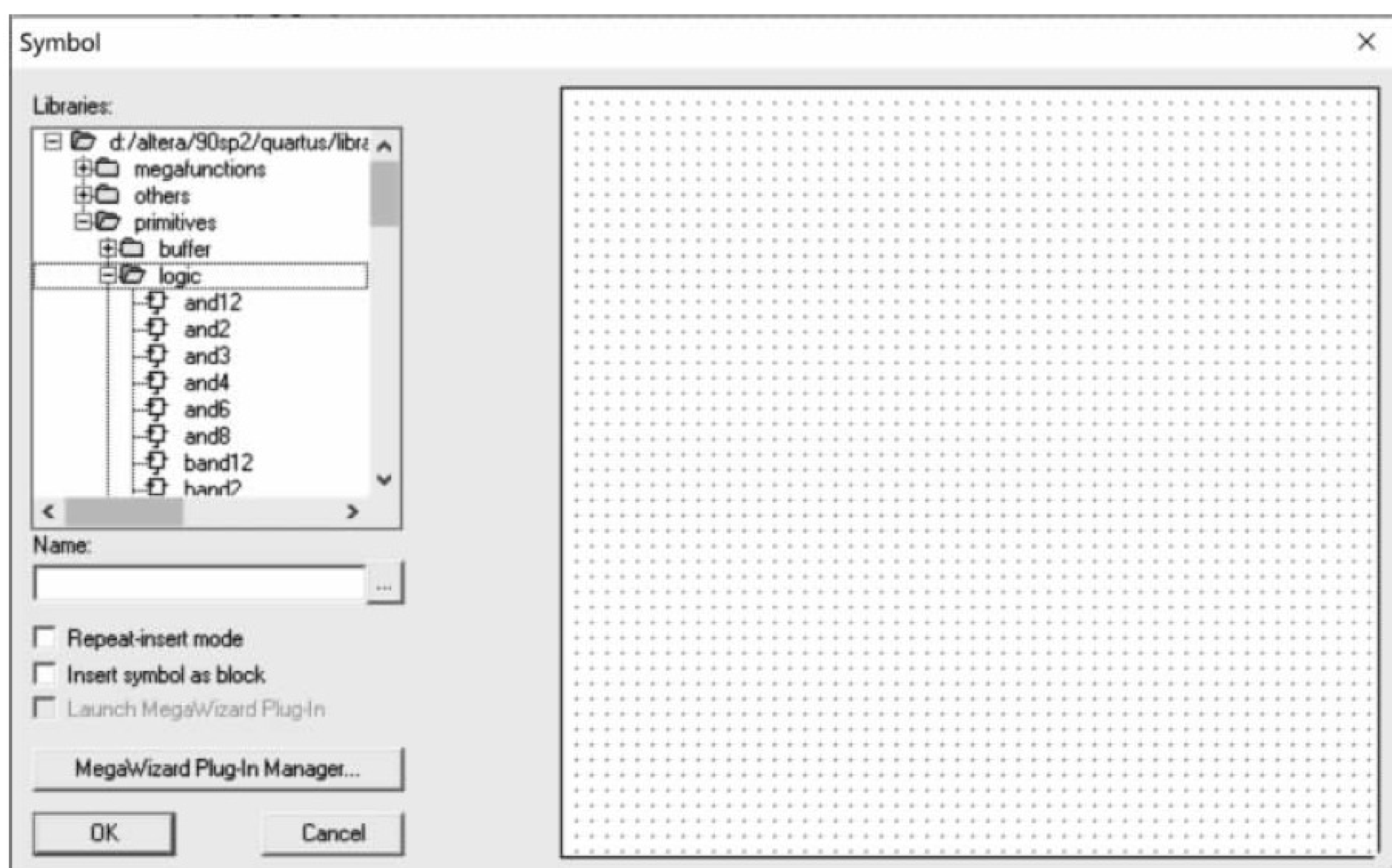


图 4.2.8　库元件选择界面

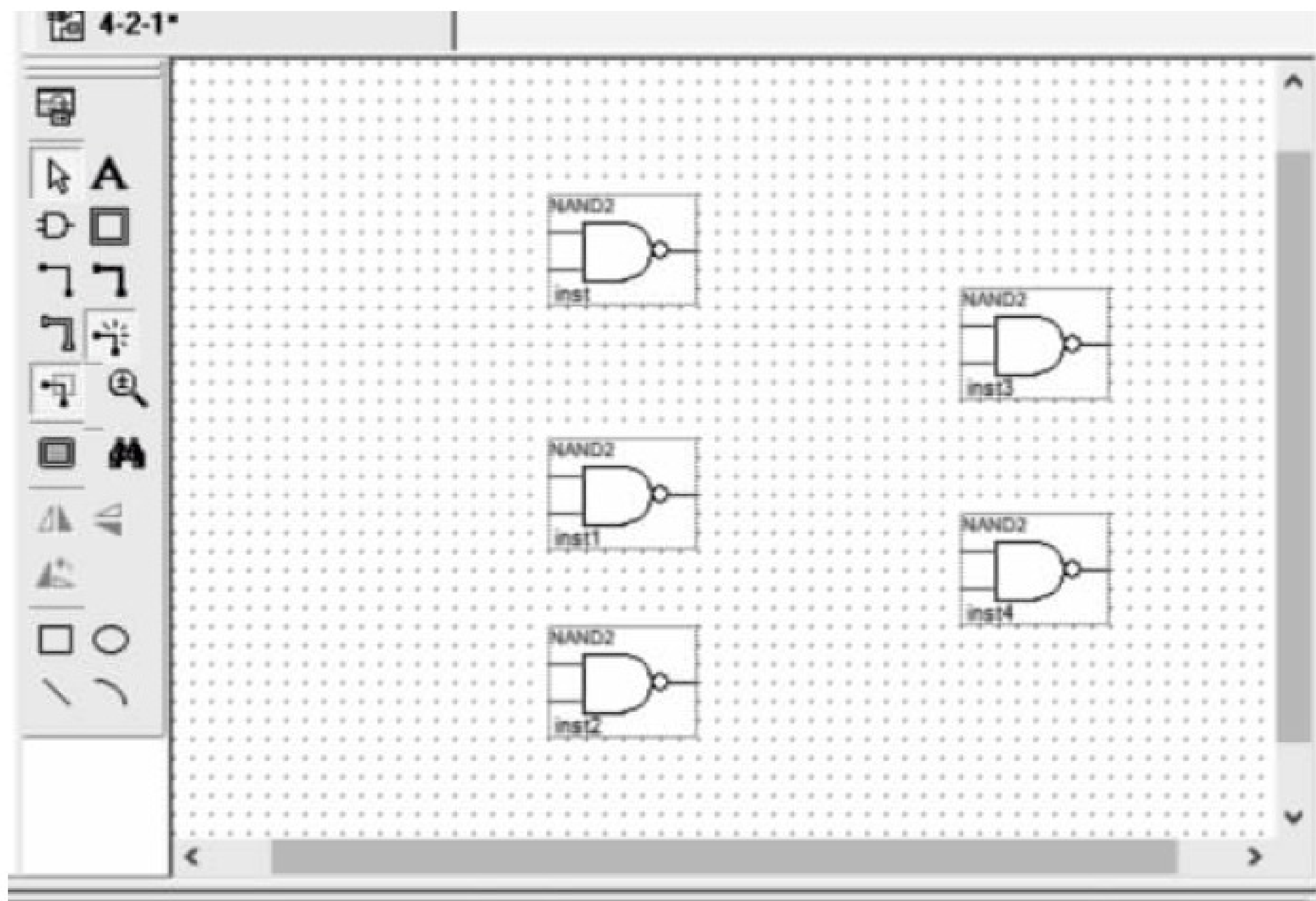


图 4.2.9　放置 5 个与非门的图

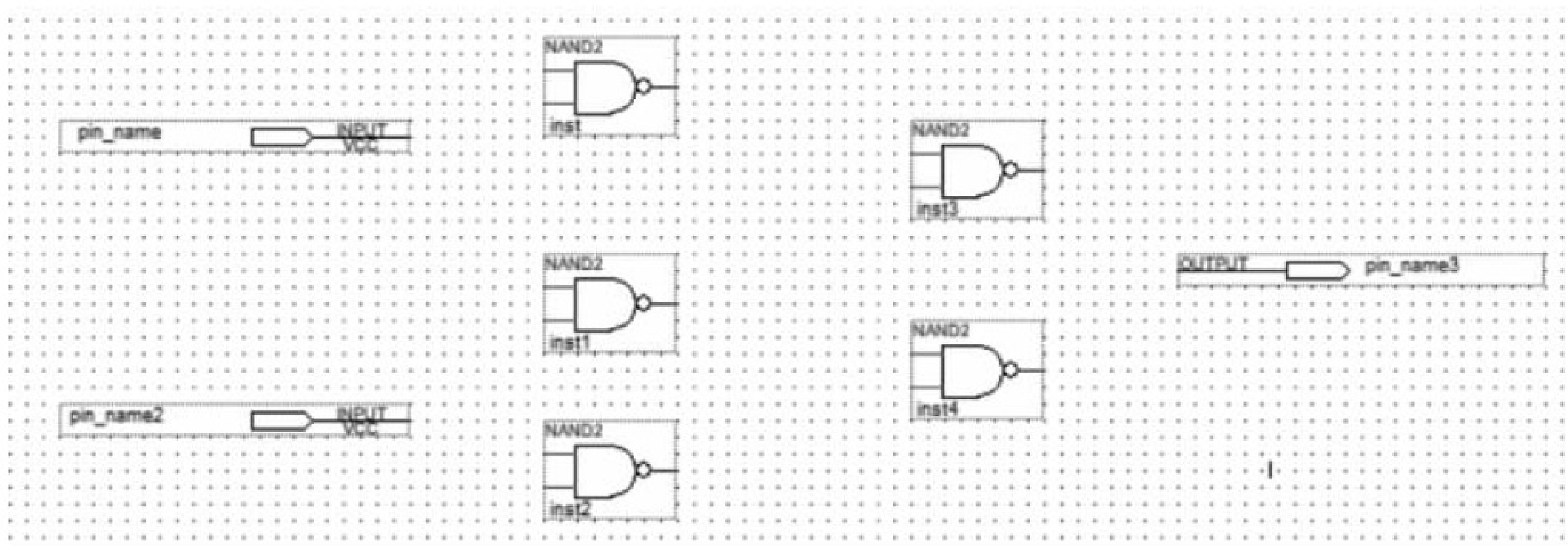


图 4.2.10　放置了全部符号的图

本框中输入 A，如图 4.2.11 所示，单击确定完成名称的修改。继续修改 pin_name2 为 B，pin_name3 为 Y。可进一步修改 5 个与非门名称为 G1～G5。

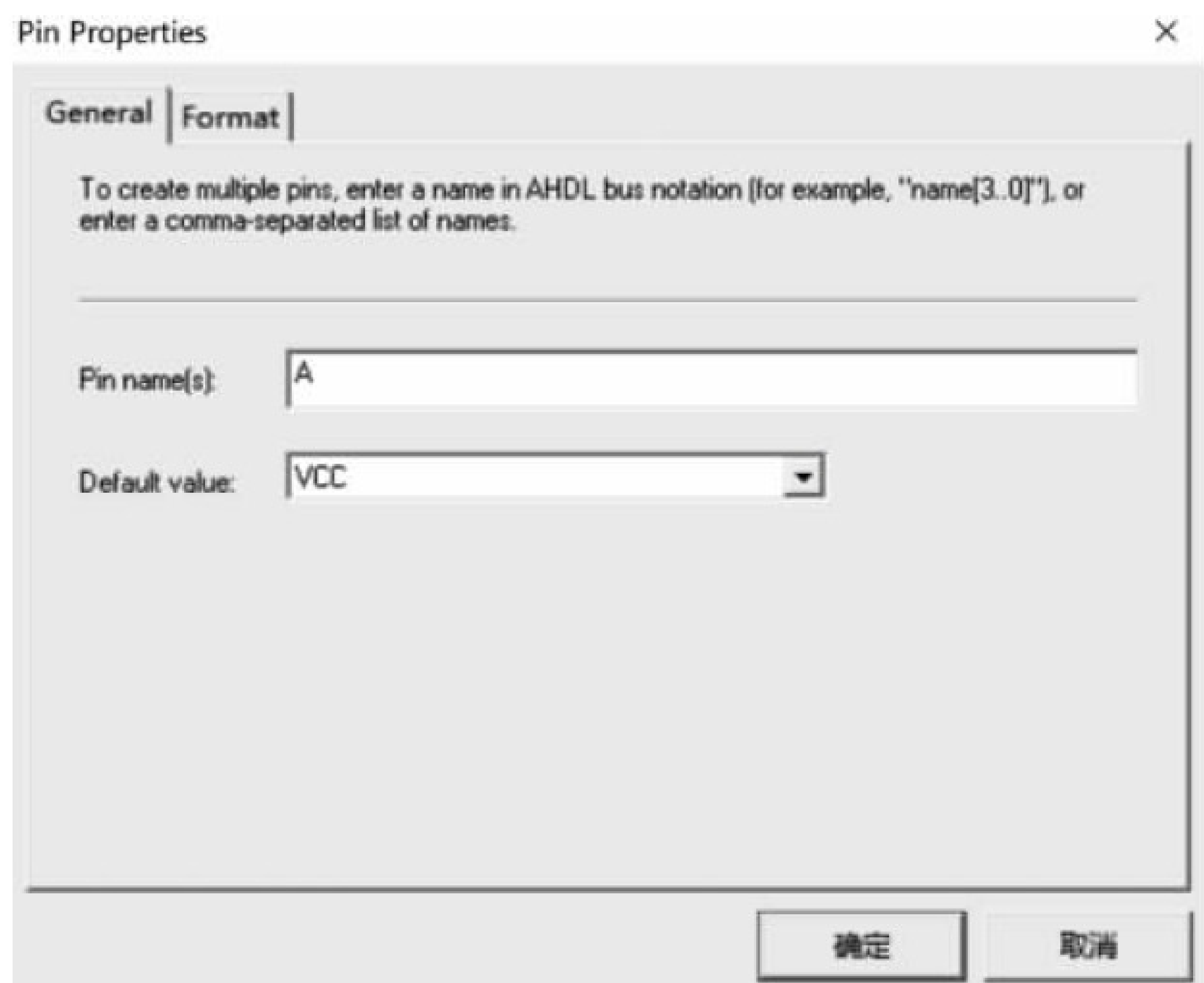


图 4.2.11　定义元件名称的图

(5) 连接电路。将鼠标指向符号引脚,若光标变为“+”,可拖放鼠标连线,依照图 4.2.1 正确连接电路。具体连线时,可在连接线的中间放开鼠标,完成局部线段的绘制。之后,将鼠标指向连接线的端点,若光标变为“+”,可继续拖放鼠标连线。

(6) 编译电路。保存设计后,选择 Processing→Start Compilation 子菜单编译电路,如果没有错误,系统将弹出编译成功消息框。

当脱离了图形编辑界面时,可选择工作区左上角的 Project Navigator 任务调板,单击调板下方的 Files 子调板,如图 4.2.12 所示。单击子调板中的具体图形文件可进入图形文件编辑界面。



图 4.2.12　Project Navigator

当编译有错误时,可选择工作区最下方的 Message 区域,适当移动最右方的滚动条,查看编译错误的提示,如图 4.2.13 所示。

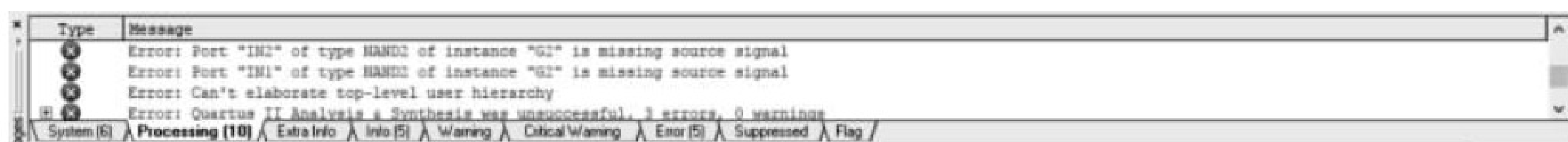


图 4.2.13　编译错误的提示 1

双击具体的错误将指出图中错误的具体位置,如图 4.2.14 所示。图中,与非门 G2 的连线没有与具体的输入元件连接,依照图 4.2.1 的要求,应与 A 连接,将鼠标指向连接线的端点,拖放鼠标完成与输入 A 的连线。

3. 建立要仿真的逻辑图形文件的波形文件

波形文件是 Quartus Ⅱ仿真的必需文件,其主要作用是定义各输入信号及要观察的输出信号。具体实现方法如下。

1) 新建波形文件并添加到当前工程中

选择 File 菜单的 New 子菜单,在弹出的任务窗口中选择文件类型为 Vector Waveform File,如图 4.2.15 所示,单击 OK 按钮进入波形文件编辑状态。选择 File 菜单的 Save As

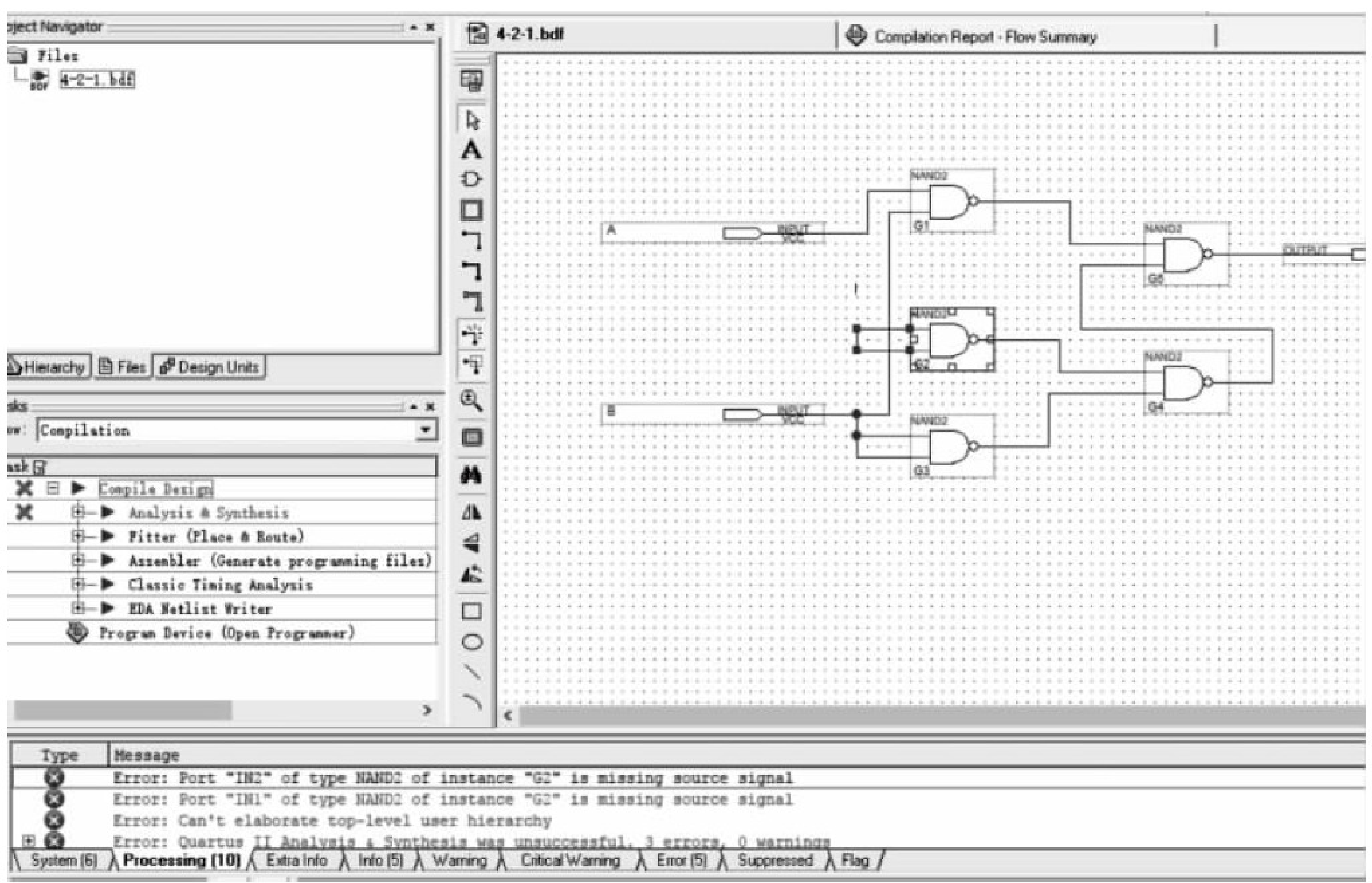


图 4.2.14　编译错误的提示 2

子菜单，将新创建的未命名的波形文件命名（默认与图形文件同名，也必须与图形文件同名），如图 4.2.16 所示，单击“保存”按钮保存。保存前确认已勾选任务窗口最下方的 Add file to current project 复选框。

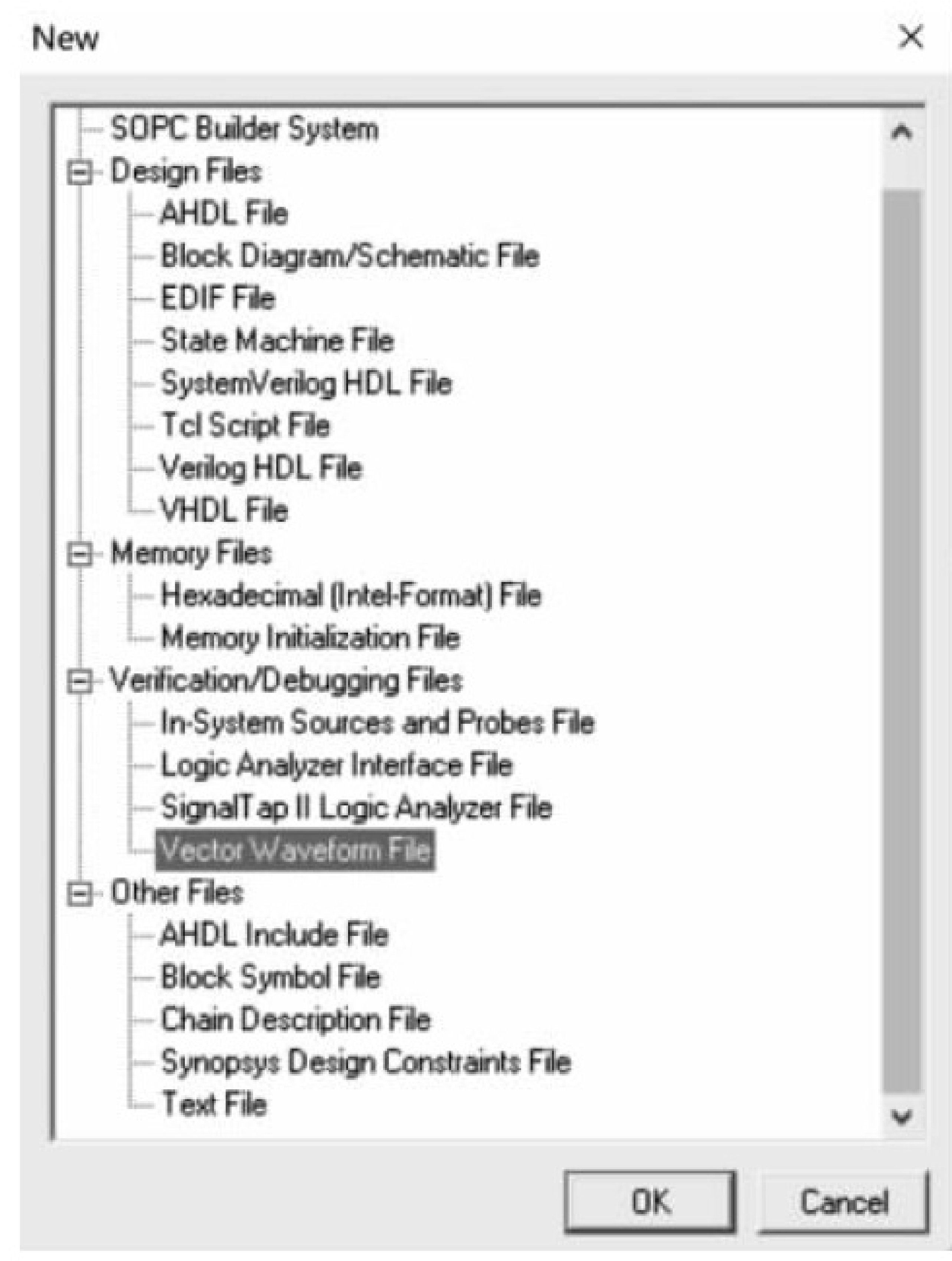


图 4.2.15　新建波形文件



图 4.2.16　另存波形文件

2）导入图形文件中定义的输入输出到波形文件中

编辑区 Name 分栏任意位置右击鼠标，在弹出的菜单中选择 Insert→Insert Node or Bus…，如图 4.2.17 所示。在随后弹出的对话框中，选择右方的 Node Finder…按钮（右方的第 3 个按钮），将出现 Node Finder 任务窗口。

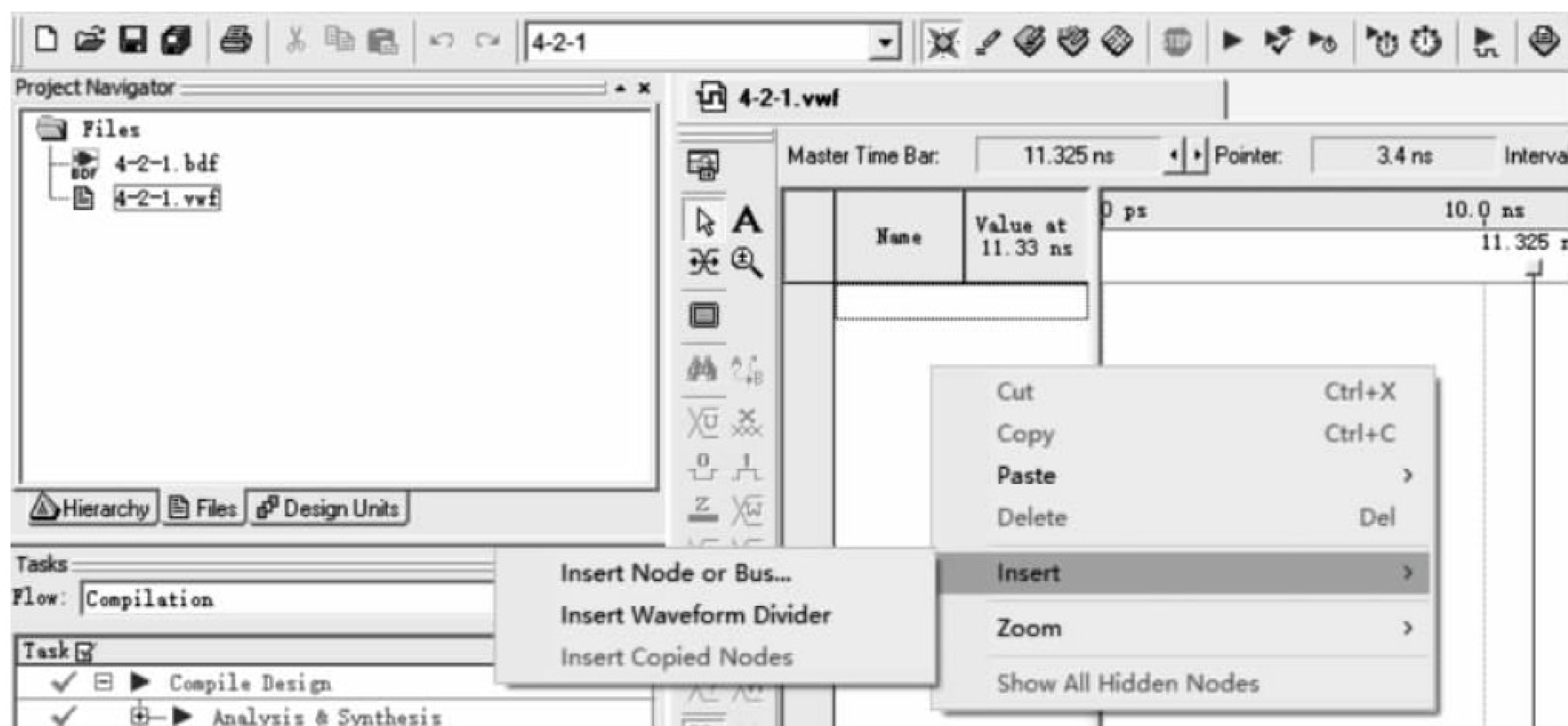


图 4.2.17　Node Finder 的图 1

在 Node Finder 任务窗口中，设置 Filter 下拉框为 Pins：unassigned，如图 4.2.18 所示。单击 List 按钮，在左下文本框中选择想要编辑或观察的信号，单击“>>”按钮，将选择的输入、输出添加到右下文本框，如图 4.2.19 所示。

单击 OK 按钮返回如图 4.2.17 所示的对话框，单击 OK 按钮返回波形编辑界面，完成图形文件中的输入输出到波形文件的导入。

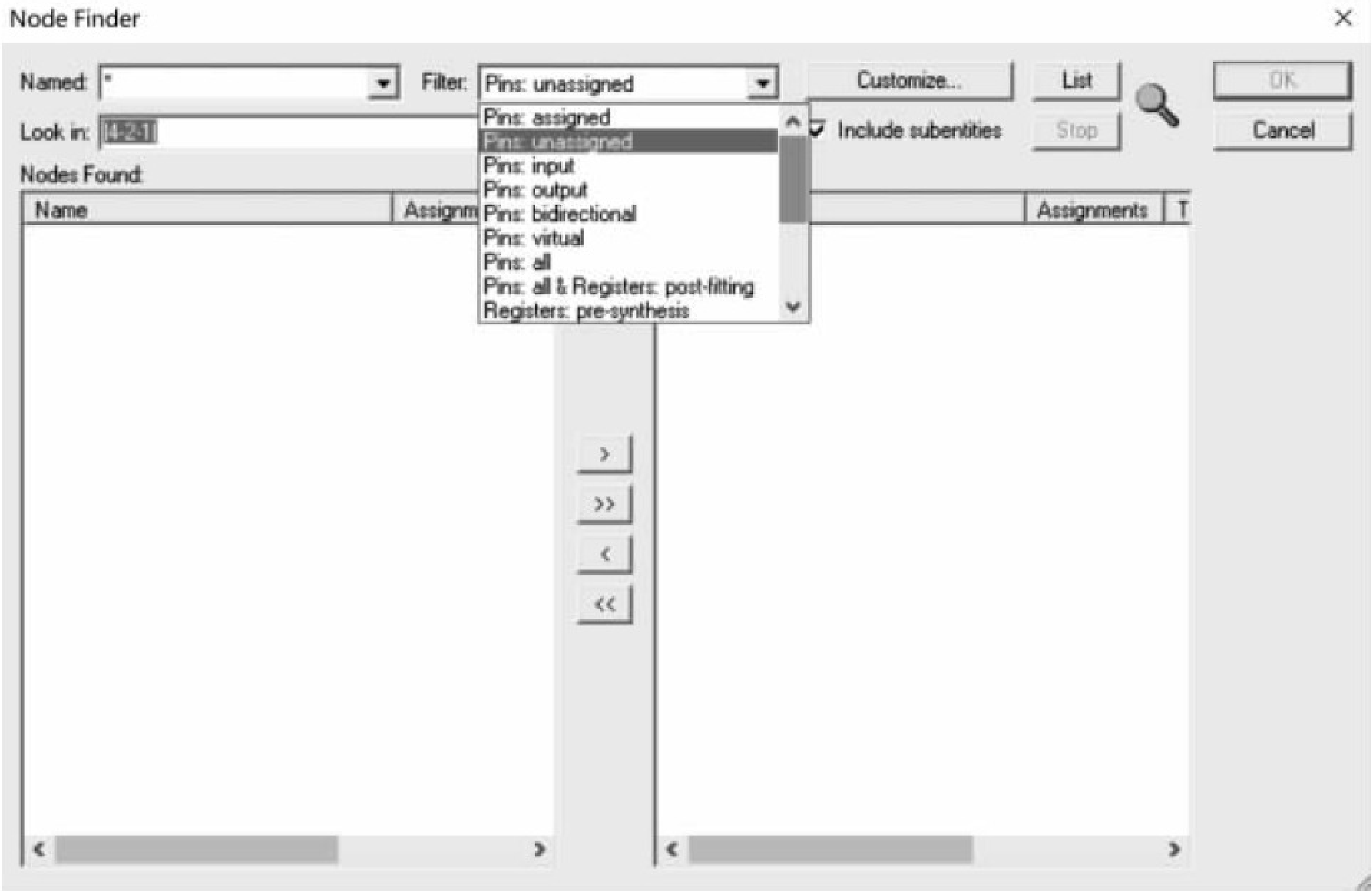


图 4.2.18 Node Finder 的图 2

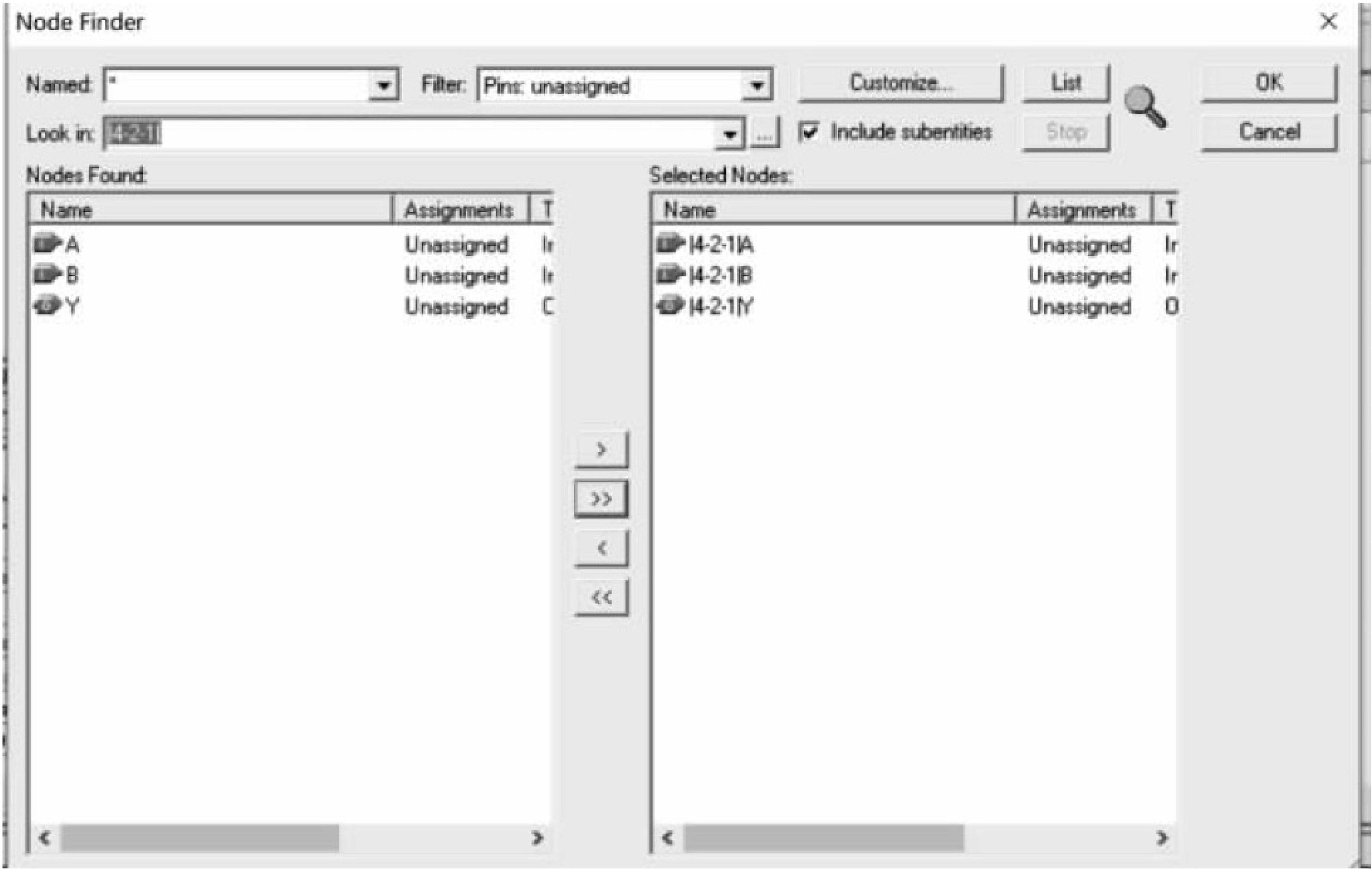


图 4.2.19 Node Finder 的图 3

3）设置输入 A、B 的值

拖动鼠标选择要设置的输入的具体的区域，单击左边工具栏中的“0”或者“1”，设置成相应的电平，如图 4.2.20 所示。图中，选择了 A、B 两个输入第 4 个时间单元，单击工具栏

中的“1”设置成高电平。

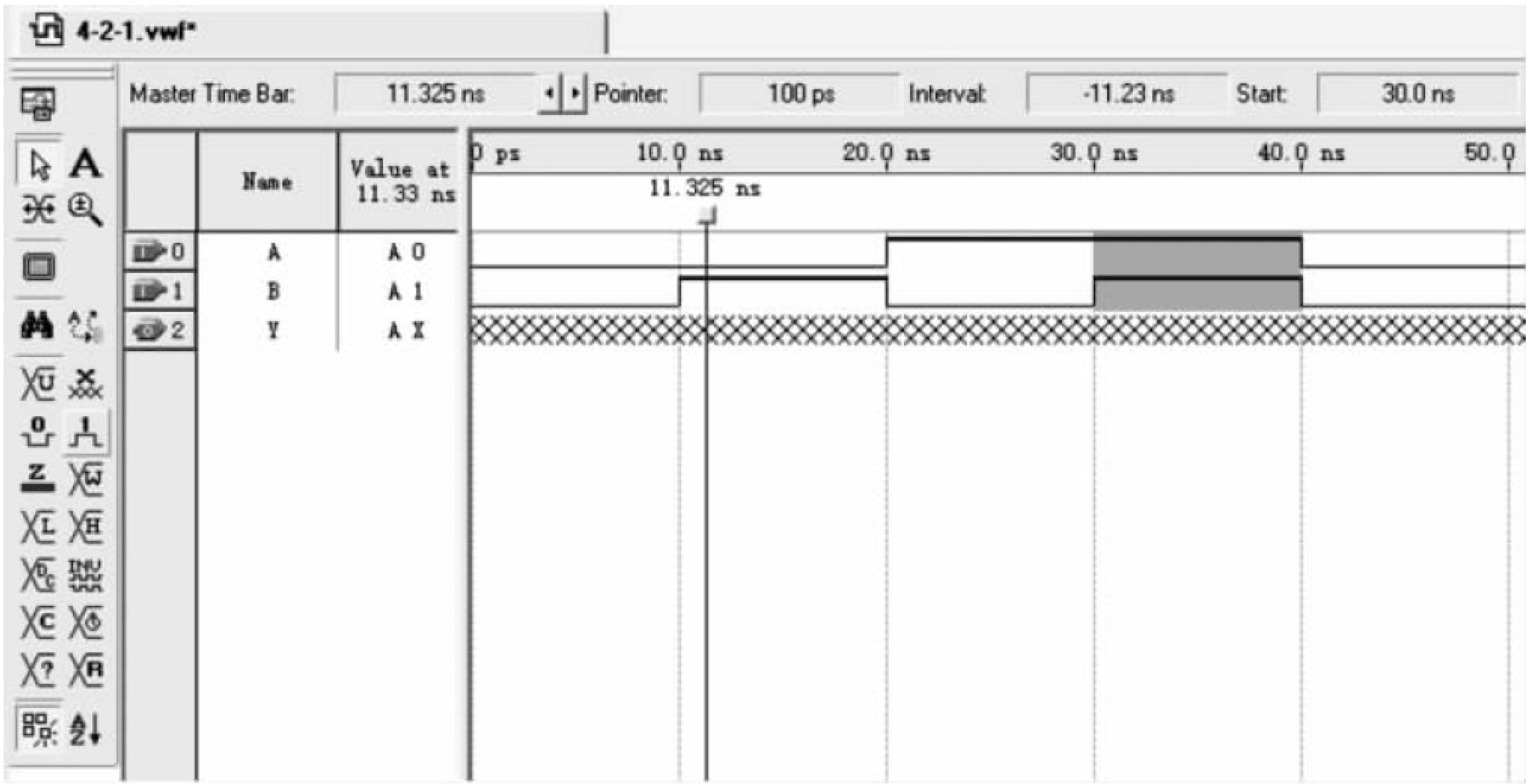


图 4.2.20　设置输入 A、B 的值

为了全方位观察电路的逻辑功能，可参考真值表的格式设置输入的值。图中，将 A、B 两个输入第 1～4 个时间单元分别设置成 00、01、10、11。

4. 仿真

保存设计后，选择 Processing→Start Simulation 子菜单(第一次仿真前需先单击菜单 Assignment 中的 Settings，在 Settings 窗的 Category 下选择 Simulator Settings，在 Simulation Mode 中选择仿真模式：时序仿真 Timing 或功能仿真 functional；在 Simulation input 中选择 xxx.vwf 仿真波形文件，然后单击 OK 按钮)，如果没有错误，系统将弹出仿真成功消息框，求出如图 4.2.2 所示的仿真波形。根据仿真波形，该仿真电路逻辑功能为同或门。

4.2.3　用小规模器件实现组合逻辑电路(SSI 设计)



显然，利用门电路等小规模器件实现组合逻辑电路的过程是分析过程的逆过程。也就是说，根据给出的实际逻辑问题，要求设计出实现这一逻辑功能的最简单的逻辑电路。这就是组合逻辑电路的设计任务。

所谓最简单的逻辑电路，是指电路所用的器件数最少，器件的种类最少，而且器件之间的连线也最少。

组合逻辑电路的设计步骤如下：

(1) 分析所提出的逻辑问题的逻辑关系，确定输入变量和输出变量。显然，应该将引起事件的原因定为输入变量，而将事件的结果定为输出变量。

(2) 根据逻辑问题的因果关系写出逻辑真值表，将以真值表形式实现的逻辑问题抽象

成逻辑函数。

(3) 由真值表写出逻辑函数表达式。

(4) 根据所选用的器件类型,将函数化简、变量转换成所需要的形式。

当使用小规模集成的门电路进行设计,例如用与非门进行设计时,为获得最简单的设计结果,应将逻辑函数式化简成最简与或式,然后将逻辑函数变换为与非的形式。

当用中规模集成的常用组合逻辑电路设计组合逻辑电路时,需要将函数式变换为适当的形式,以便能用最少的器件和最简单的连线接成所要求的逻辑电路。用中规模集成的常用组合逻辑电路设计组合逻辑电路,将在 4.3 节中介绍。

还可以使用存储器或可编程逻辑器件等大规模集成电路设计组合逻辑电路,这部分内容将在后面的章节再作讨论。

(5) 根据化简或者变换后的逻辑表达式画出最简电路。

下面通过几个例题介绍组合逻辑电路的设计方法。

【例 4.2.4】 试用与非门组成一个多数表决电路,以判别 A、B、C 三人中是否为多数赞同。

解 (1) 分析题意,写出真值表。由题意可知,该电路的输入是 A、B、C 三人的“赞同”或“反对”,输出是“多数赞同”或“多数反对”,用变量 Y 表示。对输入 A、B、C,用 0 表示“反对”,用 1 表示“赞同”;对输出,$Y=1$ 表示“多数赞同”,$Y=0$ 表示“多数反对”。据此,可列出所要设计电路的真值表如表 4.2.4 所示。

表 4.2.4　例 4.2.4 的真值表

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(2) 由表可列出 Y 的最小项表达式为

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

(3) 利用卡诺图进行化简。第(2)步的公式不为最简。将其填入卡诺图(图 4.2.21)并化简,有

$$Y = AB + BC + AC$$

A \ BC	00	01	11	10
0			1	
1		1	1	1

图 4.2.21　例 4.2.4 的图

(4) 变换为与非式,有

$$Y = \overline{\overline{AB + AC + BC}} = \overline{\overline{AC}\ \overline{BC}\ \overline{AB}}$$

(5) 将第(4)步的公式画成电路图,如图 4.2.5 所示,由例 4.2.3 可知设计正确。

【例 4.2.5】 试用与非门电路实现异或逻辑功能。

解 (1) 分析题意，写出真值表。所谓异或逻辑功能，就是指当两个输入变量(设为 A、B)不相同时输出函数 Y 为 1；否则为 0。故很容易写出如表 4.2.5 所示的真值表。

表 4.2.5 例 4.2.5 的真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

(2) 根据表 4.2.5 真值表写出逻辑函数表达式为

$$Y = A\bar{B} + \bar{A}B$$

(3) 第(2)步的公式已为最简，可变换为与非的形式，有

$$Y = A\bar{B} + \bar{A}B = \overline{\overline{A\bar{B} + \bar{A}B}} = \overline{\overline{A\bar{B}}\;\overline{\bar{A}B}}$$

(4) 进一步变换表达式。用电路实现第(3)步的公式需要 5 个与非门。从电路实现角度，该式存在进一步简化的可能，可考虑将 $\bar{A}$、$\bar{B}$ 两个与非门用一个与非门实现。

由 $A\bar{B} = A\,\overline{AB}$，有

$$Y = \overline{\overline{A\bar{B}}\;\overline{\bar{A}B}} = \overline{\overline{A\,\overline{AB}}\;\overline{\overline{AB}\,B}}$$

(5) 画电路。根据变换后的逻辑表达式 Y，画出电路如图 4.2.22 所示。

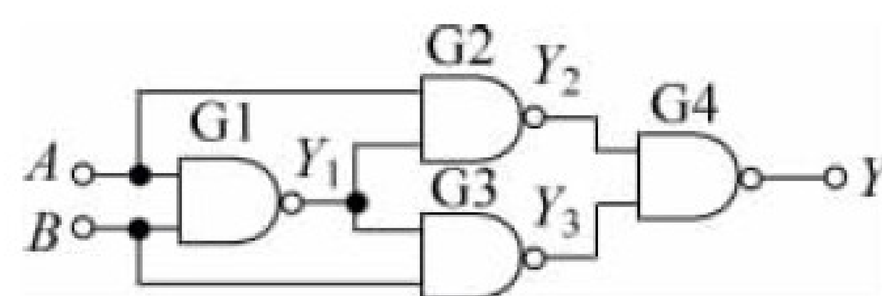


图 4.2.22 例 4.2.5 的图

【例 4.2.6】 试用与非门设计一个“三变量不一致”的判别电路。

解 (1) 定义电路的输入输出。该逻辑问题具有三个输入、一个输出。可定义电路的输入 A、B、C 和电路的输出 Y($Y=0$ 表示一致，$Y=1$ 表示不一致)。

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

$A\bar{B}$　$\bar{A}C$　$B\bar{C}$

图 4.2.23 例 4.2.6 的图

(2) 做出真值表。根据逻辑功能要求，可做出真值表如表 4.2.2 所示。

(3) 由真值表可做出卡诺图如图 4.2.23 所示。

(4) 由卡诺图写出最简逻辑函数表达式并变换为与非形式，具体如下：

$$Y = A\bar{B} + \bar{A}C + B\bar{C} = \overline{\overline{A\bar{B} + \bar{A}C + B\bar{C}}} = \overline{\overline{A\bar{B}}\;\overline{\bar{A}C}\;\overline{B\bar{C}}}$$

(5) 进一步分析与思考。用电路实现第(4)步的公式需要 7 个与非门。根据例 4.2.2 的分析可知，如图 4.2.3 所示电路只有 5 个与非门，具有和该式相同的真值表，即具有相同的逻辑功能。该电路的逻辑表达式如下：

$$Y = \overline{\overline{A\,\overline{ABC}} \cdot \overline{B\,\overline{ABC}} \cdot \overline{C\,\overline{ABC}}}$$

二者具有相同逻辑功能的理论分析过程如下：

$$
\begin{aligned}
Y &= \overline{\overline{A\overline{ABC}}\cdot\overline{B\overline{ABC}}\cdot\overline{C\overline{ABC}}} = \overline{\overline{A\overline{BC}}\cdot\overline{B\overline{AC}}\cdot\overline{C\overline{AB}}} \\
&= \overline{\overline{A(\bar{B}+\bar{C})}\cdot\overline{B(\bar{A}+\bar{C})}\cdot\overline{C(\bar{A}+\bar{B})}} \\
&= A(\bar{B}+\bar{C})+B(\bar{A}+\bar{C})+C(\bar{A}+\bar{B}) \\
&= A\bar{B}+A\bar{C}+\bar{A}B+B\bar{C}+\bar{A}C+\bar{B}C \\
&= A\bar{B}+B\bar{C}+\bar{A}C = \overline{\overline{A\bar{B}}\cdot\overline{B\bar{C}}\cdot\overline{\bar{A}C}}
\end{aligned}
$$

(6) 可画出电路如图 4.2.3 所示。可总结进一步简化电路的方法：在最简表达式的基础上，寻找有用的添加项，提取公共因子。

复习与思考

4.2.1　试利用与非门组成与门、或门、非门、或非门。

4.2.2　试写出如图 4.2.24 所示电路输出信号的逻辑表达式，并判断能否化简。若能，请化简并用与非门实现。

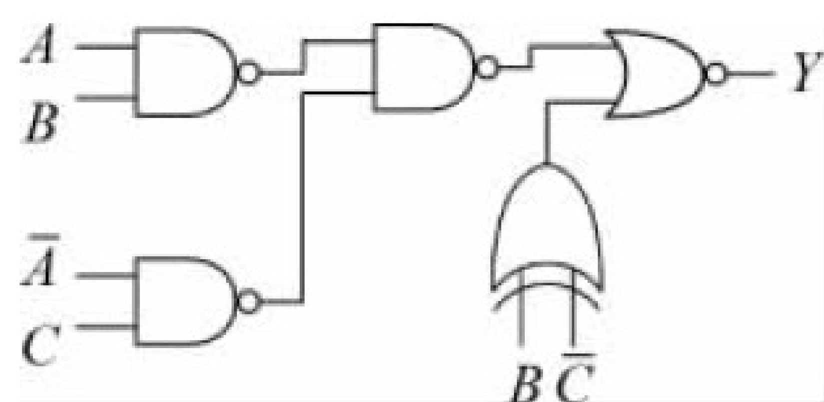


图 4.2.24　复习与思考 4.2.2 的图

4.3　常见组合逻辑电路的逻辑特点及其应用

人们发现，在实践中为解决各种逻辑问题而设计出的逻辑电路，其中有些逻辑电路经常、大量地出现在各种数字系统当中。这些电路包括编码器、译码器、数据选择器、数值比较器、加法器、函数发生器、奇偶校验器/发生器等。为了使用方便，厂家将这些逻辑电路制成了中、小规模集成的标准化产品。

4.3.1　编码器



一般地讲，用文字、符号或者数字表示特定对象的过程称为编码。例如，孩子出生时家长给取名字，开运动会时给运动员编号等，都属于编码。不过，前者是用汉字进行编码，后者是十进制数进行编码。在数字电路中，为了区分一系列不同的事物，将其中的每个事物用一系列逻辑 0 和逻辑 1 按一定规律编排起来，组成不同的代码来表示，这就是编码。

在数字电路中，信号都是以高、低电平的形式给出的。编码就是把输入的高、低电平形式的信号编成一个对应的二进制代码。执行编码功能的电路统称为编码器。

根据编码器输入是否有优先级，编码器分为普通编码器和优先编码器两类。每类中，又包括二进制编码器、二—十进制编码器两种类型。用 n 位二进制代码对 $N=2^n$ 个信号进行编码的电路称为二进制编码器，实现二—十进制编码的电路称为二—十进制编码器。

1. 二进制编码器

根据上面的定义，n 位的二进制编码器输出为 n 个二进制位，输入为 2^n 个信号。

3 位的二进制编码器框图如图 4.3.1 所示。图中，输出为 3 根线，输入为 8 根线，常形象地将 3 位二进制编码器描述为 8 线-3 线编码器(简称 8-3 编码器)。

2. 二—十进制编码器

二—十进制编码器能将 10 个输入信号分别编成 10 个 BCD 码，每个 BCD 码用 4 个二进制位表示，框图如图 4.3.2 所示。可见，二—十进制编码器输出为 4 根线，输入为 10 根线，常形象地将二—十进制编码器描述为 10 线-4 线编码器(简称 10-4 编码器)。

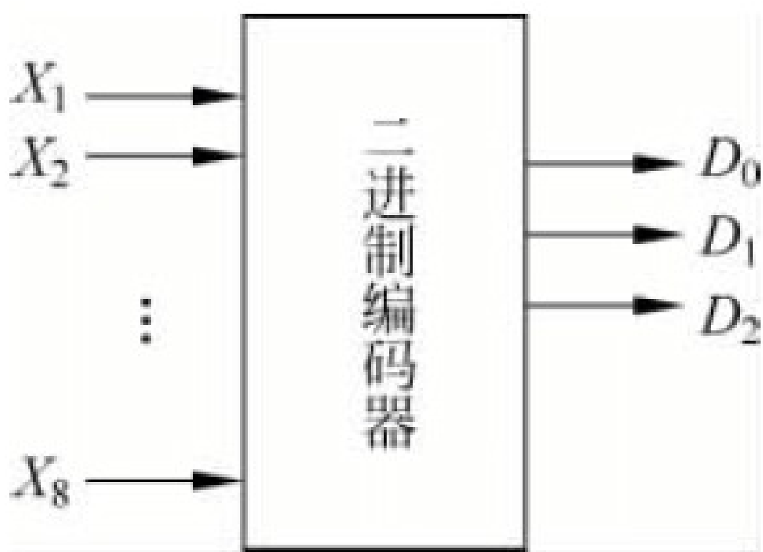


图 4.3.1　二进制编码器

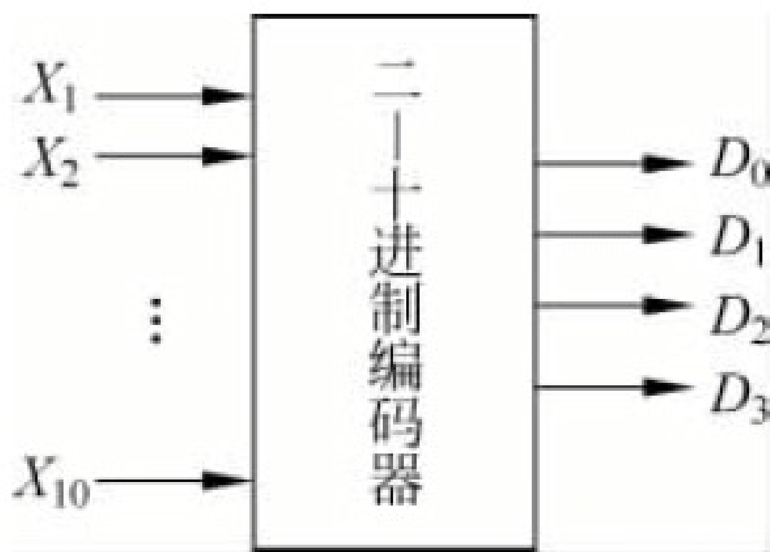


图 4.3.2　二—十进制编码器

必须指出，从编码角度，二进制编码器、二—十进制编码器并无本质区别，可对照如图 4.3.3 所示的 4 位二进制编码器来进一步理解。

4 位二进制编码器具有 4 个输出、16 个输入。二—十进制编码器具有 4 个输出、10 个输入。可见，对输入不多于 10 个的编码应用，二者均可完成该编码应用。

当然，从扩展角度，二进制编码器具有扩展功能，两个 4 位的二进制编码器可构成一个 5 位的二进制编码器。二—十进制编码器因为输出所具有的编码信息有余量(不够才需要扩展)，因此不方便扩展，集成的二—十进制编码器也没有扩展功能，两个二—十进制编码器只是两个独立的编码器，不能构成一个二—二十进制的编码器。

3. 普通编码器的逻辑特点



由如图 4.3.1～图 4.3.3 所示的框图可以看出，编码器在任一时刻，可完成且只能完成一个特定输入的编码。当多个输入信号同时到达时，只能完成其中的一个输入编码。能对多个输入按照优先级排序，只对其中优先级最高的输入编码，具有这样功能的编码器称为优先编码器。普通编码器不能实现多个输入优先级的排序。

当然，不管是优先编码器还是普通编码器，均可完成对输入的编码，具有编码电路的逻辑特点，下面结合如图 4.3.4 所示的 3 位二进制普通编码器进行分析。

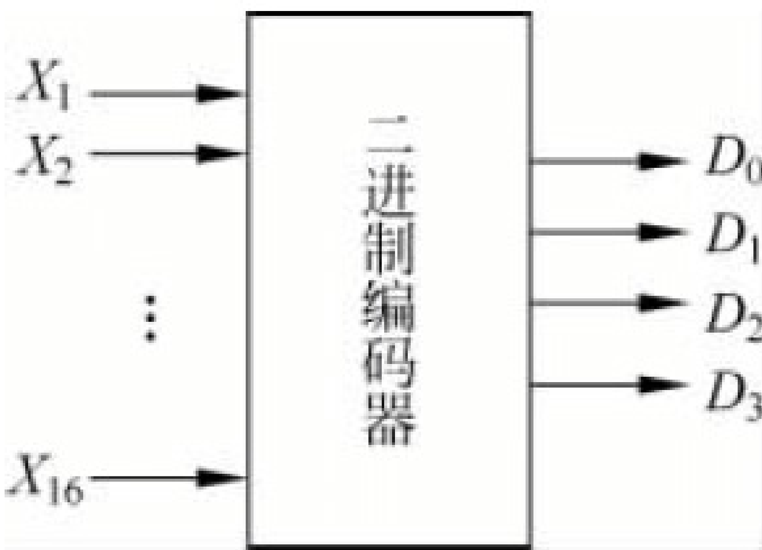


图 4.3.3　4 位二进制编码器

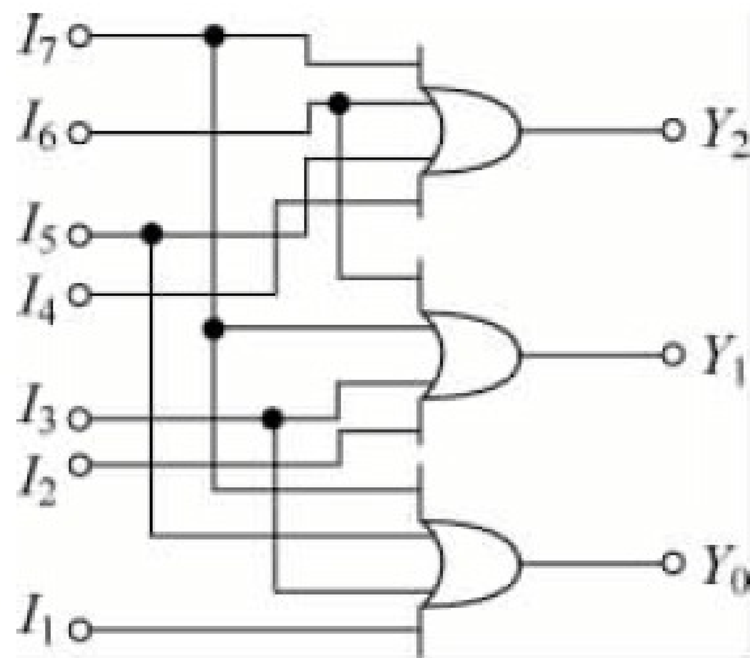


图 4.3.4　3 位二进制普通编码器

将如图 4.3.4 所示的普通编码器电路输入到 Quartus Ⅱ中编译并仿真，得到的波形如图 4.3.5 所示。

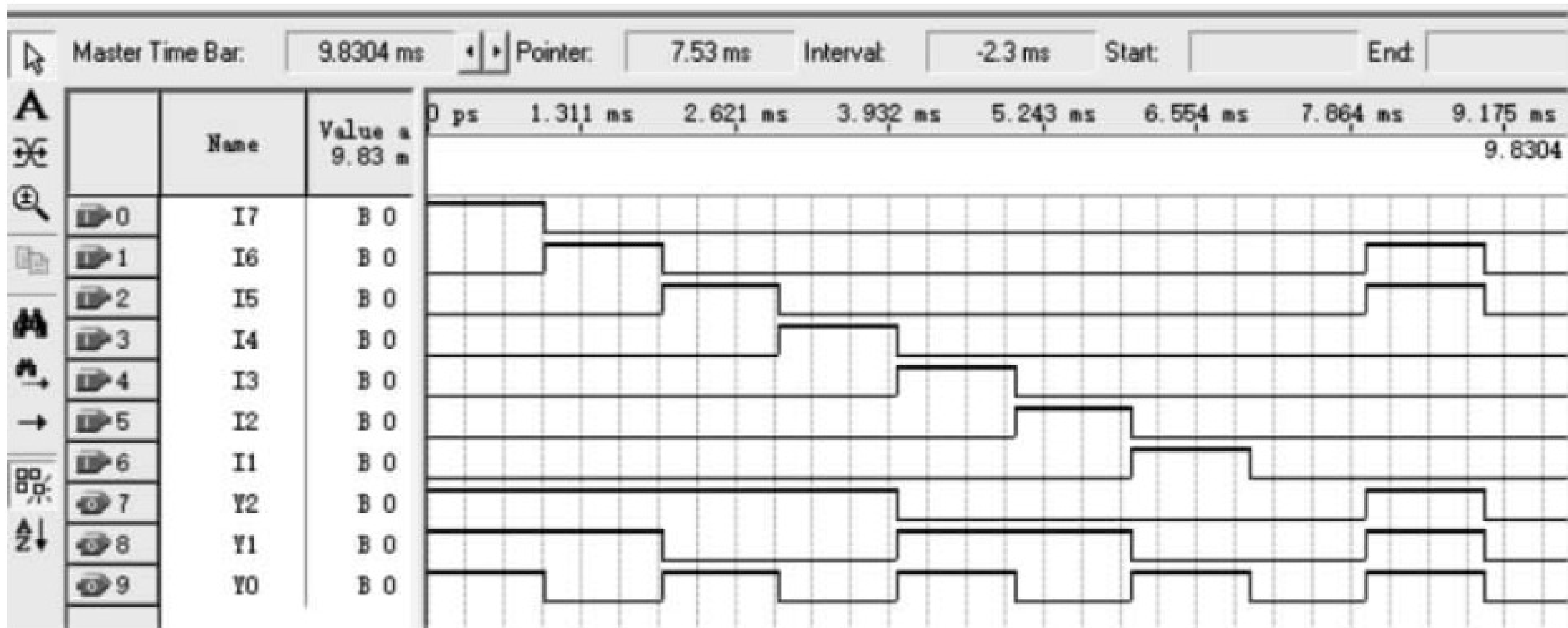


图 4.3.5　3 位二进制普通编码器仿真波形

由波形的前 8 个单元可得图 4.3.4 所示编码器的真值表，如表 4.3.1 所示。

表 4.3.1　如图 4.3.4 所示编码器的真值表

I_1	I_2	I_3	I_4	I_5	I_6	I_7	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	1	1
0	0	0	1	0	0	0	1	0	0
0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	1	1	1	1

如果规定，在任一时刻只能有一路输入端有信号到来，其余输入端均无信号到达，有信号用 1 表示，无信号用 0 表示，则表 4.3.1 所示的逻辑电路可完成 8 路输入信号的编码(全 0 为 I_0输入有效)。8 路输入信号 I_0、I_1、I_2、I_3、I_4、I_5、I_6、I_7编码分别为 000、001、010、011、100、101、110、111。

如图 4.3.4 所示的 3 位普通编码器的逻辑功能特点：可完成 8 路输入信号的编码，将每个有效的输入信号(1 为有效)编码为对应的 BCD 码。如 I_5有效，则输出值为 5 的 BCD 码"101"。

参考 3 位普通编码器的逻辑功能特点，对照如图 4.3.2 所示的二—十进制编码器的框图，可类推出二—十进制编码器的逻辑功能特点：可完成 10 路输入信号的编码，每个有效的输入信号编码为对应的 8421BCD 码。如 I_8有效，则输出值为 8 的 BCD 码"1000"。

当然，也可采用逻辑代数的分析方法分析如图 4.3.4 示的 3 位普通编码器的逻辑功能，由电路可写出各个输出函数表达式如下：

$$\begin{cases} Y_2 = I_4 + I_5 + I_6 + I_7 \\ Y_1 = I_2 + I_3 + I_6 + I_7 \\ Y_0 = I_1 + I_3 + I_5 + I_7 \end{cases} \tag{4.3.1}$$

规定任一时刻只能有一路输入端有信号到来，其余输入端均无信号到达。有信号用 1 表示，无信号用 0 表示。当 I_4、I_5、I_6、I_7 分别为 1 时，$Y_2=1$，对应真值表 4.3.1 的第 4～8 列。可类似分析 Y_1、Y_0，从而求出如表 4.3.1 所示的真值表。

编码器任一时刻只能完成多个输入中的一个输入的编码，而普通编码器不具有对多个输入同时有效时的优先排序功能，因此，应用实践中，普通编码器必须遵循任一时刻只能有一路输入端有信号到来的约束。细心的读者观察如图 4.3.5 所示的仿真波形时可能注意到：波形的第 9 个单元输入信号 I_5、I_6 同时有效，输出为全 1，编码器的输出发生混乱。

普通编码器可用于人工场合下少量输入的编码，相对缺乏实用性，解决的方法是采用优先编码器。



4. 优先编码器及其应用

可通过集成芯片二—十进制优先编码器 74LS147[①] 来理解优先编码器的逻辑功能。上面介绍的普通编码器输入为 1 有效，输出为对应输入的 8421BCD 码。

74LS147 优先编码器增加了优先编码功能，各输入信号按照 $\bar{I}_9$、$\bar{I}_8$、$\bar{I}_7$、$\bar{I}_6$、$\bar{I}_5$、$\bar{I}_4$、$\bar{I}_3$、$\bar{I}_2$、$\bar{I}_1$ 优先级逐渐降低。为方便应用，以低电平输入为有效信号，输出为十进制数码对应 8421BCD 码的反码。

根据上面的描述，可写出 74LS147 优先编码器的真值表如表 4.3.2 所示。由真值表可看出，当 $\bar{I}_8$、$\bar{I}_5$、$\bar{I}_3$ 同时有效时，只对 $\bar{I}_8$ 编码，输出值为 8 的 8421BCD 码的反码“0111”。若 $\bar{I}_2$、$\bar{I}_6$ 同时为 0，其余输入均为 1，只对 $\bar{I}_6$ 编码，编码输出为“0110”的反码，即“1001”。

表 4.3.2 74LS147 优先编码器的真值表

$\bar{I}_1$	$\bar{I}_2$	$\bar{I}_3$	$\bar{I}_4$	$\bar{I}_5$	$\bar{I}_6$	$\bar{I}_7$	$\bar{I}_8$	$\bar{I}_9$	$\bar{Y}_3\bar{Y}_2\bar{Y}_1\bar{Y}_0$
1	1	1	1	1	1	1	1	1	1111
0	1	1	1	1	1	1	1	1	1110
×	0	1	1	1	1	1	1	1	1101
×	×	0	1	1	1	1	1	1	1100
×	×	×	0	1	1	1	1	1	1011
×	×	×	×	0	1	1	1	1	1010
×	×	×	×	×	0	1	1	1	1001
×	×	×	×	×	×	0	1	1	1000
×	×	×	×	×	×	×	0	1	0111
×	×	×	×	×	×	×	×	0	0110

74LS147 为常用芯片，其引脚说明如图 4.3.6 所示。紧靠四边形的小圆圈表示“低电平为有效信号”。四边形内部标注为引脚功能说明。四边形外部标注为引脚编号。如右上表

① 数字集成电路芯片主要有 CMOS、TTL 两大系列，各系列相同序号的芯片逻辑功能相同。TTL 的常用系列有 74LS 系列等；CMOS 的 74HC 系列已在很多场合下取代 74LS 系列；本书配有完备的视频，为与视频中的系列一致，纸质教材统一使用 74LS 系列芯片，读者可结合第 3 章的内容理解不同系列在应用实践中的差异。

示芯片第16脚为电源。

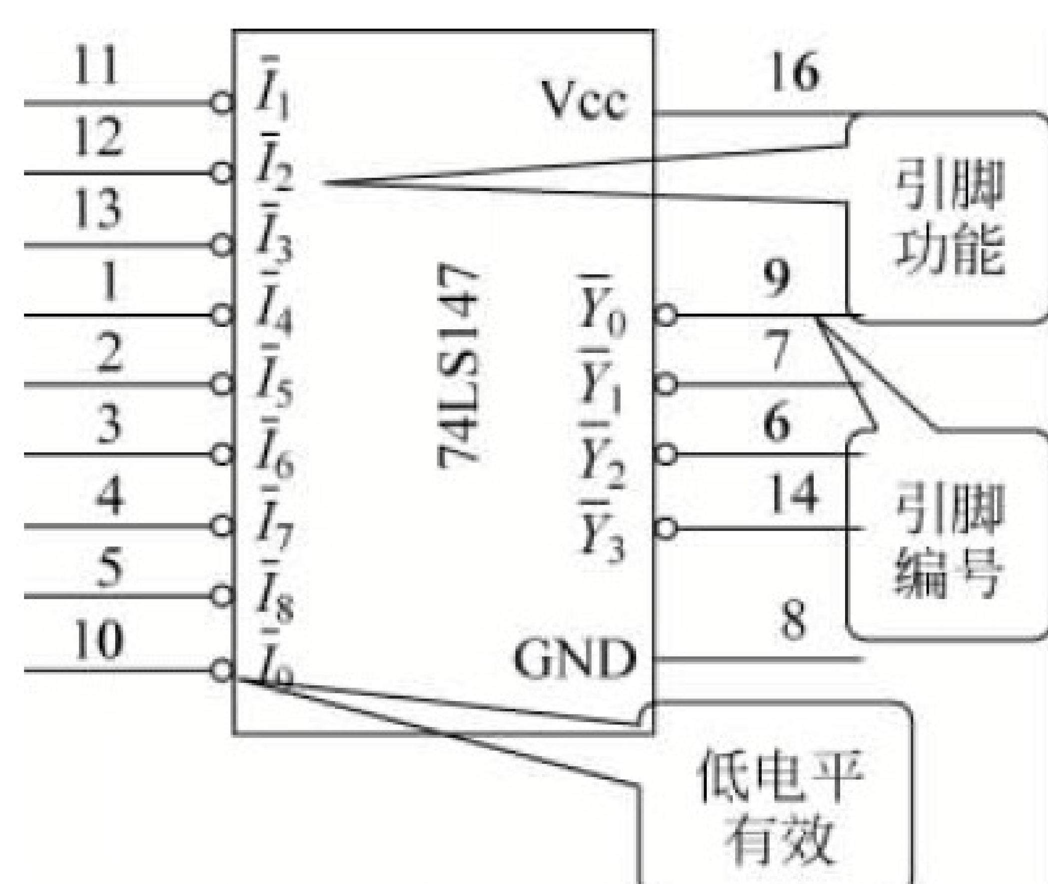


图 4.3.6　74LS147 引脚说明

为便于读者绘制电路图，在本书中，芯片引脚顺序没有采用实际引脚顺序，实际芯片引脚编号方法如图 4.3.7 所示(16 引脚两列直插芯片)。可从引脚功能图直接得出 74LS147 的逻辑图，如图 4.3.8[①] 所示。

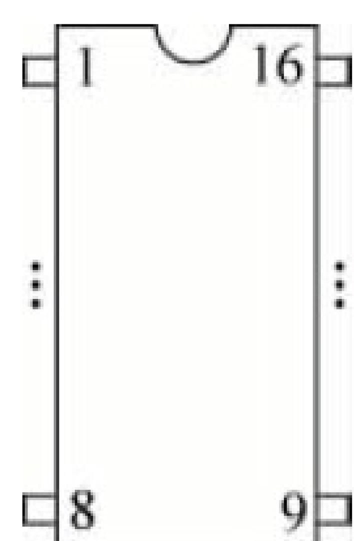


图 4.3.7　引脚编号方法

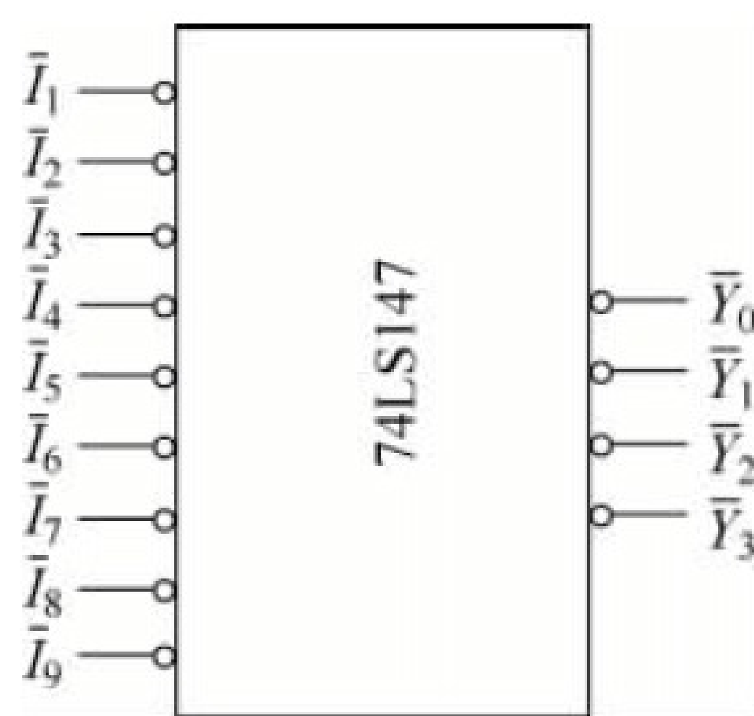


图 4.3.8　逻辑符号

如图 4.3.9 所示为 QuartusⅡ中库元件 74147 的原理图。1N～9N 对应 9 个外部编码输入，“没有输入”对应 0N；AN～DN 为 4 位二进制 BCD 码输出。读者可参考该电路理解优先编码器优先级的电路实现方法。如 8N、9N 优先级最高，二者取反后经或非门接到了所有多输入与门的 1 个输入端。当 8N、9N 中有 1 个输出为 0 时，或非门输出为 0，所有多输入与门的输入被封锁，确保了 8N、9N 的优先级。读者也可以进一步分析该电路，分析如何确保 9N 的优先级高于 8N 的。

二进制优先编码器与二—十进制优先编码器在原理上并无本质区别，但考虑二进制优先编码器的扩展，增加了相应的控制及扩展控制位。图 4.3.10 所示为 3 位二进制优先编码器 74LS148 的引脚图。$\bar{S}_T$、$\bar{Y}_{EX}$、Y_S为控制引脚，解释如下：

(1) $\bar{S}_T$ 为选通输入端。当 $\bar{S}_T=0$ 时允许编码，芯片工作；当 $\bar{S}_T=1$ 时，输入、输出及控制引脚 $\bar{Y}_{EX}$、Y_S均被封锁，编码被禁止。

---

① 引脚图的内部标注为单纯的引脚功能说明。逻辑图的标注说明应吻合电路的逻辑功能。如外部输入“$\bar{I}_1$”，当该引脚标注在框里时应标注为“I_1”。对集成芯片，本书使用引脚图绘制该芯片。在由该芯片构成的应用电路中，为求简洁，没有绘制引脚编号。

图 4.3.9 QuartusⅡ中库元件 74147 原理图

$\bar{S}_T$ 为芯片的选通输入端，常简称“片选”。“片选”的一个重要作用是“选片”。当多个芯片协同工作时，可使某个芯片的“片选”为 0，其余的芯片“片选”为 1，为 0 的芯片工作，其余的芯片不工作，从而实现选片的功能。基于上面的功能，常把“片选”称为使能端。

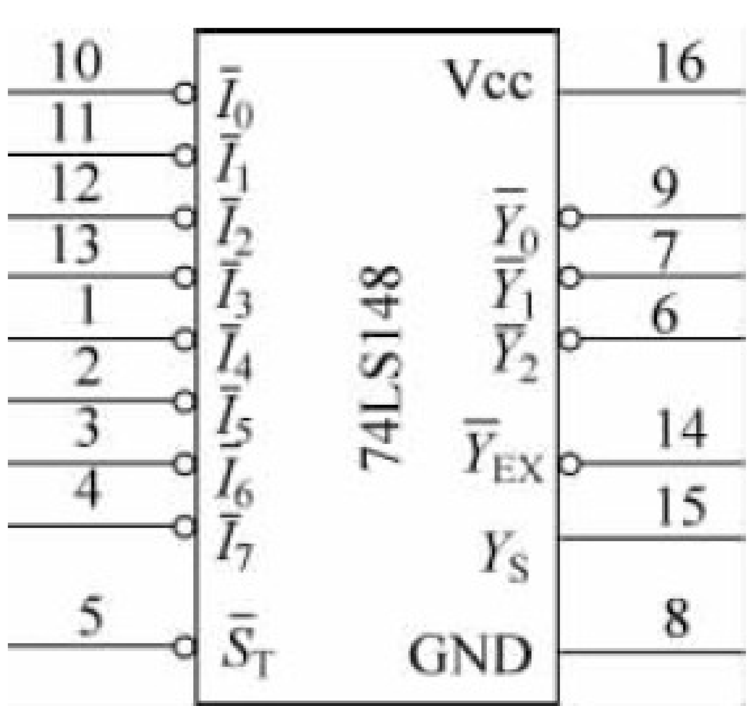


图 4.3.10 74LS148

(2) Y_S 是选通输出端。$Y_S=0$ 表示“电路工作，但无输入信号”。

可简单地将该引脚的功能理解为“选通其他芯片的输出端”。当然，要选通其他芯片，应使该芯片的“片选”为 0，使该芯片工作。依照多个芯片协同工作的应用要求，某芯片工作时，其余芯片应不工作。Y_S接另一个芯片的片选，当 $Y_S=0$ 时，芯片本身无输入信号，无须编码；与该引脚连接的另一个编码器片选有效，工作，正常编码。确保两个编码芯片中某时刻最多有一个芯片在执行编码功能。

(3) $\bar{Y}_{EX}$为扩展输出端。$\bar{Y}_{EX}=0$ 表示“电路工作，而且有输入信号”。

根据上面的逻辑功能，级联应用时，高位芯片的 Y_S端与低位芯片的 $\bar{S}_T$ 端连接起来，高

位芯片的 $\overline{Y}_{EX}$ 可作为高位的编码输出位。

当然，芯片的逻辑功能是通过电路来实现的。如图 4.3.11 所示为 QuartusⅡ中库元件 74148 原理图。

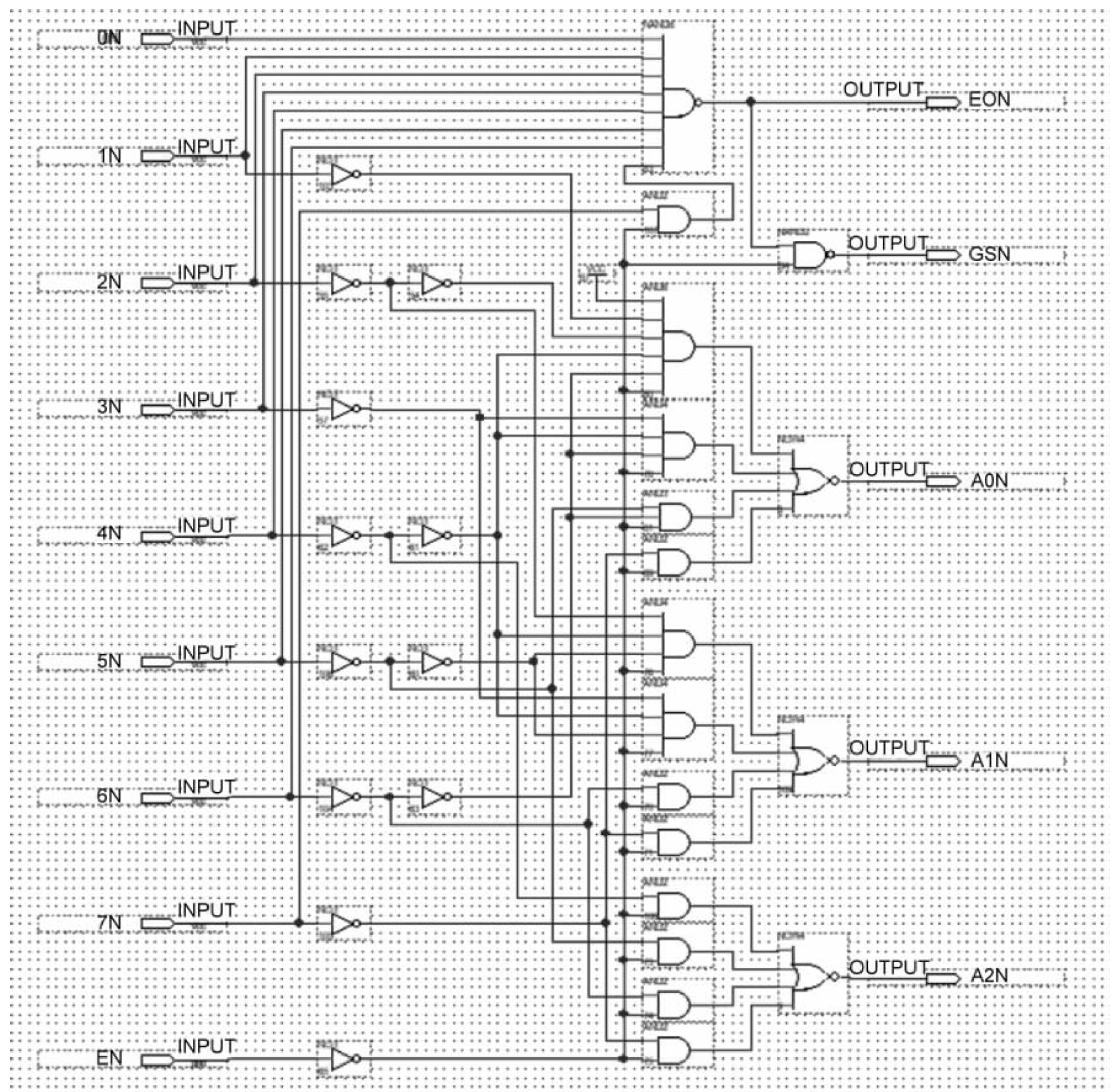


图 4.3.11　QuartusⅡ中库元件 74148 原理图

图 4.3.11 中，0N～7N 对应 8 个外部编码输入。每个外部输入经反相器接入编码电路，保证了“低电平”为有效输入的应用要求。A0N～A2N 为 8 个外部编码输入对应的 3 位二进制 BCD 码输出。每个外部输出均取反后输出，确保反码输出的逻辑功能。

EN 对应芯片的片选 $\overline{S}_T$，EON 对应芯片的选通输出端 Y_S。GSN 对应芯片的扩展输出端 $\overline{Y}_{EX}$。有兴趣的读者可参考该电路进一步理解该 3 个外部扩展引脚的逻辑功能。

74LS148 的输出为对应输入信号二进制码的反码。优先级为 $\overline{I}_7$～$\overline{I}_0$ 逐渐降低。对照 74LS147，可写出 74LS148 优先编码器真值表，如表 4.3.3 所示。

表 4.3.3 74LS148 优先编码器真值表

$\overline{S}_T$	$\overline{I}_0$	$\overline{I}_1$	$\overline{I}_2$	$\overline{I}_3$	$\overline{I}_4$	$\overline{I}_5$	$\overline{I}_6$	$\overline{I}_7$	$\overline{Y}_{EX}$	Y_S	$\overline{Y}_2\overline{Y}_1\overline{Y}_0$
1	×	×	×	×	×	×	×	×	1	1	111
0	1	1	1	1	1	1	1	1	1	0	111
0	0	1	1	1	1	1	1	1	0	1	111
0	×	0	1	1	1	1	1	1	0	1	110
0	×	×	0	1	1	1	1	1	0	1	101
0	×	×	×	0	1	1	1	1	0	1	100
0	×	×	×	×	0	1	1	1	0	1	011
0	×	×	×	×	×	0	1	1	0	1	010
0	×	×	×	×	×	×	0	1	0	1	001
0	×	×	×	×	×	×	×	0	0	1	000

【例 4.3.1】 试用两片 74LS148 接成 16-4 优先编码器，将 $\overline{A}_0$～$\overline{A}_{15}$ 16 个低电平输入信号编为 0000～1111 的 16 个 4 位二进制代码。其中 $\overline{A}_{15}$的优先权最高，$\overline{A}_0$ 的优先权最低。



解 由于 74LS148 是 8-3 优先编码器，它只有 8 个编码输入。所以，应选用两片 74LS148 优先编码器，将 16 个编码输入信号分别接到两片 74LS148，具体接法如图 4.3.12 所示。

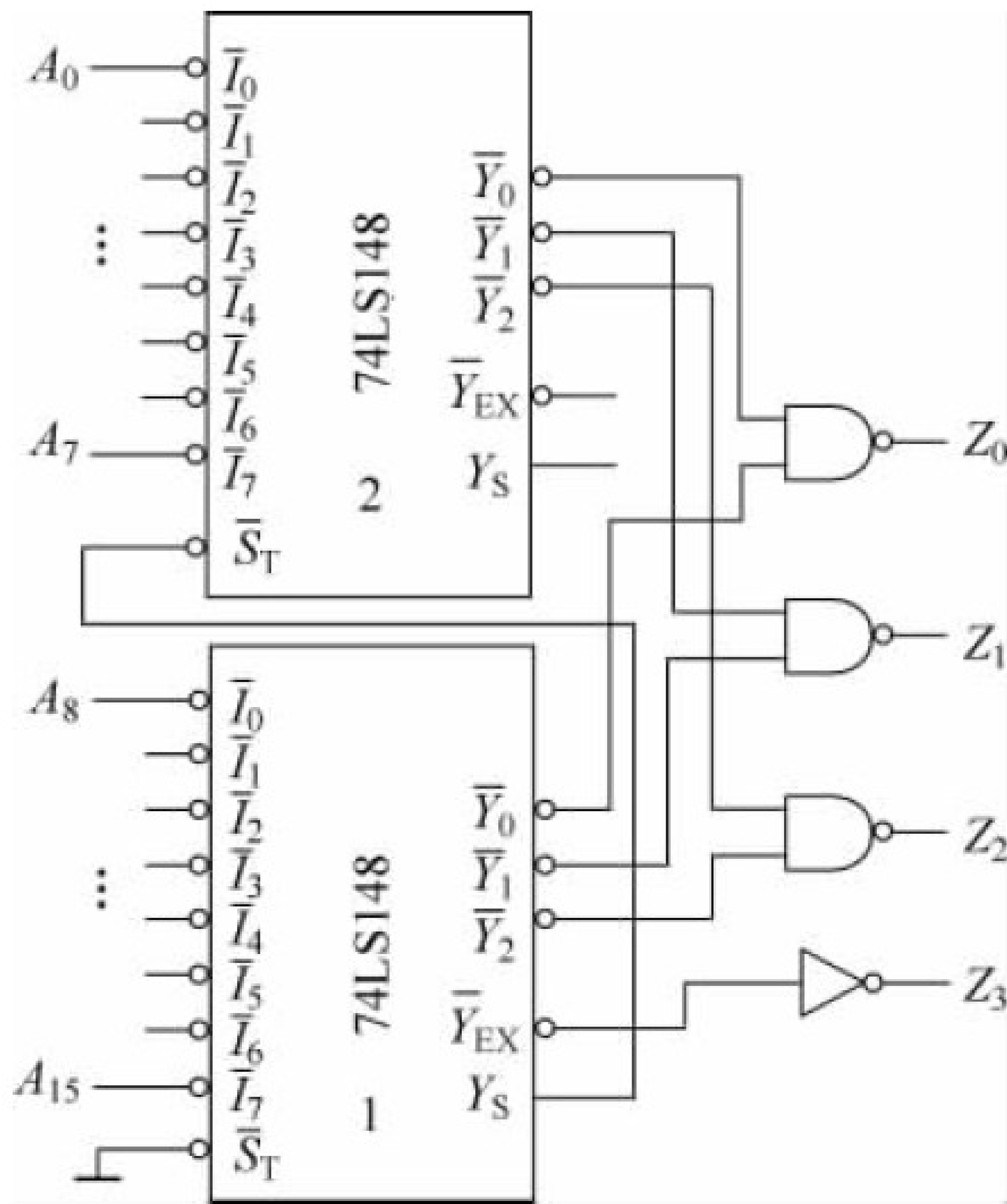


图 4.3.12 例 4.3.2 的电路图

简要解释如下：

(1) 优先级设计。两片 74LS148 具有 16 个编码输入，和题中要求吻合，只需完成优先级的设计即可完成输入的设计。

假定片 1 高于片 2，可将 $\overline{A}_8$～$\overline{A}_{15}$ 8 个优先权高的输入信号接到片 1 的 $\overline{I}_0$～$\overline{I}_7$ 输入端，而将 $\overline{A}_0$～$\overline{A}_7$ 8 个优先权低的输入信号接到片 2 的 $\overline{I}_0$～$\overline{I}_7$ 输入端。

(2) 优先级实现。按照优先顺序的要求，片 1 高于片 2，只有 $\overline{A}_8$～$\overline{A}_{15}$均无输入信号时，

才可对 $\overline{A}_0 \sim \overline{A}_7$ 的输入信号进行编码。为此，可把片 1 的"无编码信号输入"信号 Y_S 作为片 2 的选通输入信号 $\overline{S}_T$ 来保证优先顺序的要求。

(3) 低 3 位输出的设计与实现。依照该电路输入的连接特点，当片 1 编码时，低 3 位的输出为片 1 的输出；当片 2 编码时，低 3 位的输出为片 2 的输出。

上述问题可进一步转化为逻辑问题：

A 有效，$Y=A$；B 有效，$Y=B$。求 Y。

下面直接给出该逻辑问题的答案：

当高电平为有效输入时，$Y=A+B$；当低电平为有效输入时，$Y=AB$。

因此，编码输出的低 3 位为 2 片 74LS148 的输出对应位的与非（注意 74LS148 编码输出为反码，而本题要求输出原码）。

(4) 最高位输出的实现。最高位 Z_3 的逻辑功能如下：

当片 1 工作且有输入时，片 1 编码，编码值为 8～15，最高位输出为 1；当片 1 工作无输入时，片 2 编码，编码值为 0～7，最高位输出为 0。

当片 1 有编码信号输入时，它的 $\overline{Y}_{EX}=0$；无编码信号输入时，它的 $\overline{Y}_{EX}=1$，正好可以用它取反后作为输出编码的第 4 位。

同理，可用 4 片 74LS148 接成 32-5 优先编码器，以此类推。



【例 4.3.2】 某医院有 1、2、3、4 号病房 4 间，装有 4 个呼叫器，对应的护士室有 1、2、3、4 号 4 个指示灯。优先级按照 1、2、3、4 的顺序降低设置，请设计该控制电路。

解　选用 74LS148 结合门电路实现。

(1) 先实现优先级设计。设呼叫器按钮按下时输出低电平，可令 $\overline{I}_7$、$\overline{I}_6$、$\overline{I}_5$、$\overline{I}_3$ 分别对应 1、2、3、4 号病房的 4 个呼叫器输入。

(2) 求出指示灯函数 L_4、L_3、L_2、L_1。设输出为 1 时灯亮，可列出 L_4、L_3、L_2、L_1 真值表如表 4.3.4 所示。

表 4.3.4　例 4.3.2 的真值表

$\overline{I}_3$	$\overline{I}_5$	$\overline{I}_6$	$\overline{I}_7$	Y_S	$\overline{Y}_2\ \overline{Y}_1\ \overline{Y}_0$	$L_4\ L_3\ L_2\ L_1$
1	1	1	1	0	1 1 1	0 0 0 0
0	1	1	1	1	1 0 0	1 0 0 0
×	0	1	1	1	0 1 0	0 1 0 0
×	×	0	1	1	0 0 1	0 0 1 0
×	×	×	0	1	0 0 0	0 0 0 1

由真值表，可求出各指示灯函数 L_4、L_3、L_2、L_1 如下：

$$L_4 = \overline{Y}_2 Y_S, \quad L_3 = \overline{Y}_1 Y_S$$

$$L_2 = \overline{Y}_0 Y_S, \quad L_1 = \overline{\overline{I}}_7$$

可画出电路如图 4.3.13 所示。

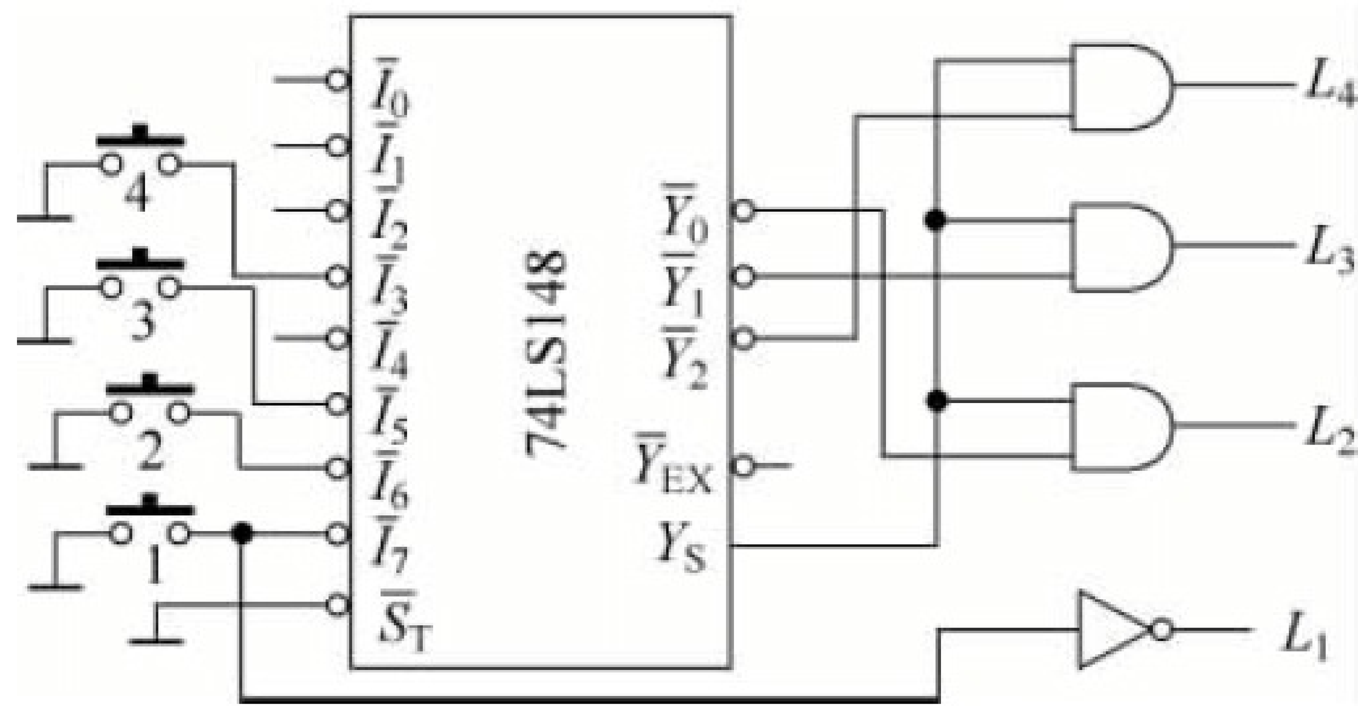


图 4.3.13　例 4.3.2 的电路图

4.3.2　译码器



1. 译码器的概念与种类

编码时，对每一种二进制代码状态，都赋予了特定的含义，即都表示了一个确定的信号或者对象。译码是编码的逆过程，实现译码操作的电路称为译码器。

如图 4.3.14(a)所示为 8-3 编码器，可从译码为编码逆过程的角度进一步理解译码器的逻辑功能。从输入、输出角度，相应的译码器应该具有 3 个输入、8 个输出，常形象地将这样的译码器描述为 3-8 译码器。此外，编码器实现了对每个确定有效的输入相应地输出一组确定的代码，如确定 X_4 为有效输入，将输出 4 的二进制 BCD 码或者其反码。从逆过程角度，对译码器而言，给定一组确定含义的代码，将获得一个有效输出。如给定一组输入 $A_2A_1A_0$ 为“100”，将获得一个有效输出 Y_4。

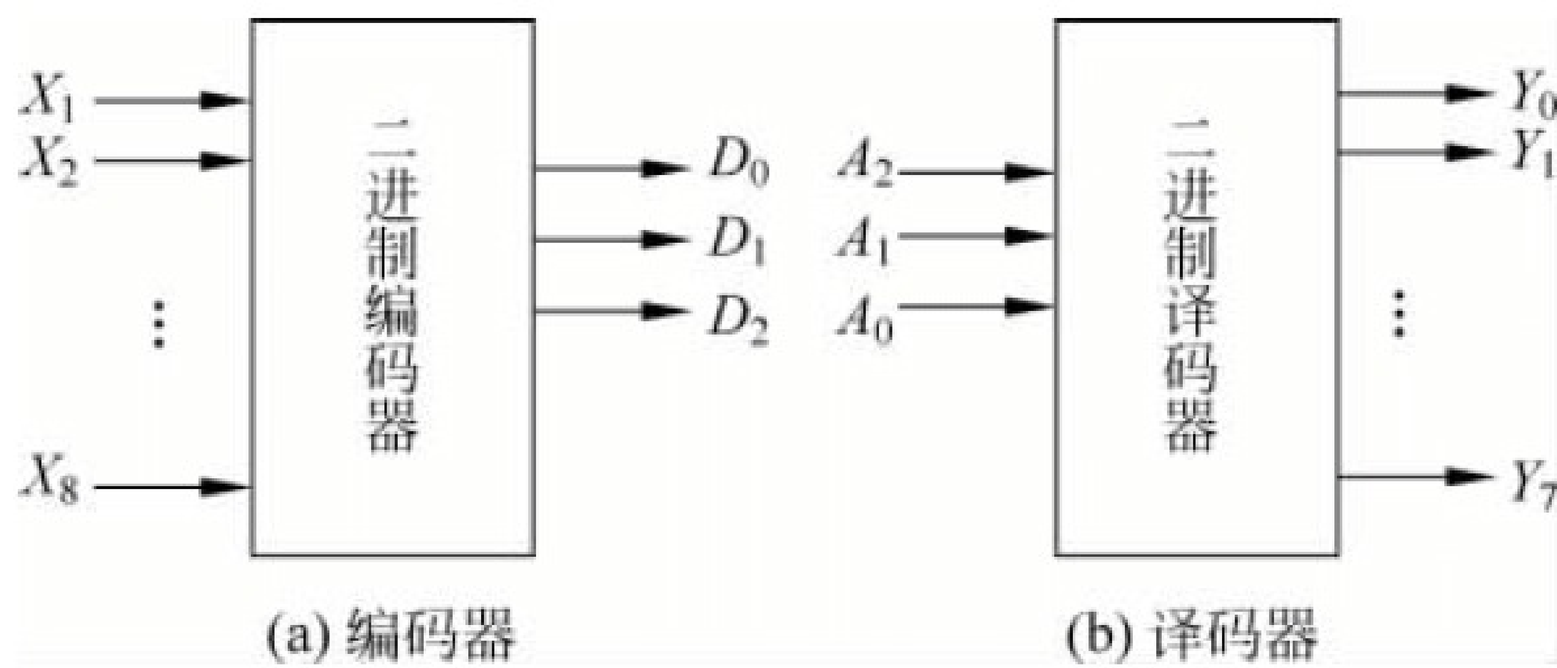


图 4.3.14　编码器、译码器对照框图

译码器的特点是可以将输入二进制代码的状态翻译成输出信号，以表示原来含义。将代码状态的特点含义“翻译”出来的过程称为译码。

应用实践中，译码器包括二进制译码器、二—十进制译码器和数字显示译码器三种典型电路。

将 n 位二进制输入码翻译为 $N=2^n$ 个输出中对应的一个有效输出的电路称为二进制译码器。实现二—十进制译码的电路称为二—十进制译码器。将输入的数码翻译成数码管、液晶等数字显示设备需要的代码的电路称为数字显示译码器，将在后面的小节中具体介绍。

由二进制译码器的定义可知，4-16 译码器具有 4 个输入、16 个输出，框图如图 4.3.15(a)所示。类似地，二—十进制译码器输出有 10 个，相对应的输入必须有 4 个二进制位，框图如图 4.3.15(b)所示。

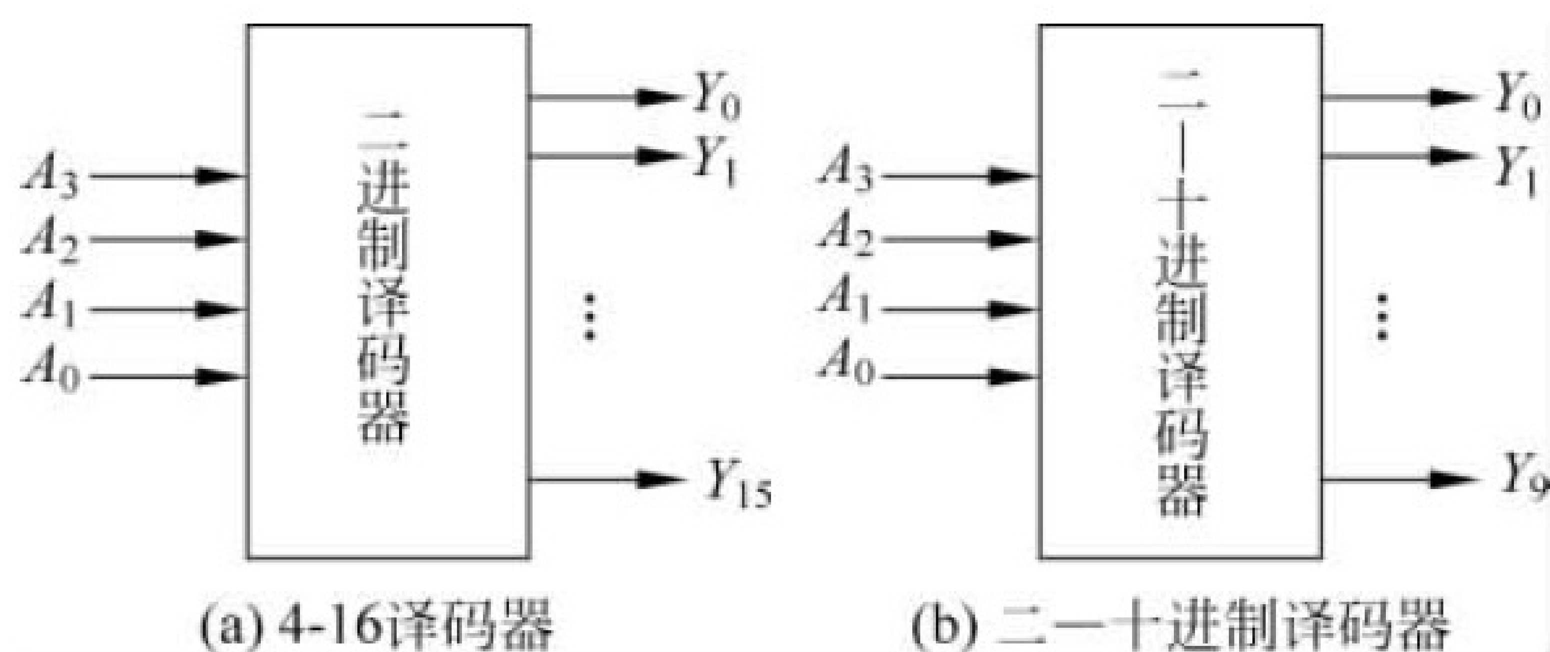


图 4.3.15　二进制译码器、二—十进制译码器对照框图

从如图 4.3.15 所示框图不难看出，二进制、二—十进制译码器并无本质区别，对输出不多于 10 个的译码应用，二者可以互换。二者的主要区别在于，二—十进制译码器只有 10 个输出，不支持扩展。



2. 二进制译码器的逻辑功能

下面结合如图 4.3.16 所示的 2-4 译码器介绍二进制译码器的逻辑功能。图中，输入代码为 2 个二进制位，输出为 4 个(设低电平为有效输出)。依照二进制译码器的定义，给定 A_1A_0 为“00”，0 路输出有效，输出为 0(低电平有效)，其余 3 个输出为 1，写成表达式如下：

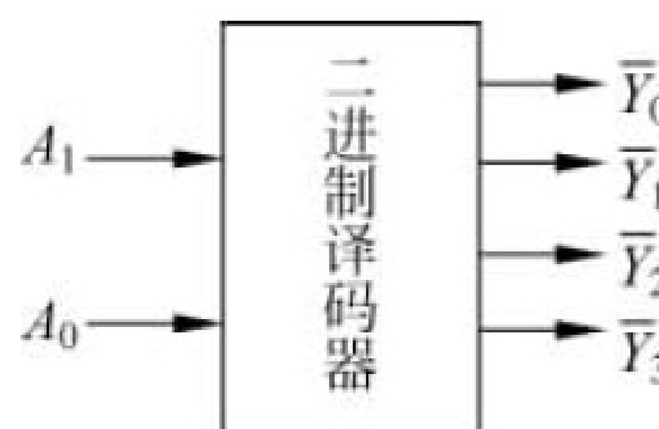


图 4.3.16　2-4 译码器

$$\overline{Y}_0 = \overline{\overline{A}_1\overline{A}_0} = \overline{m}_0$$

类似地，A_1A_0 为“01”时，1 路输出有效；A_1A_0 为“10”时，2 路输出有效；A_1A_0 为“11”时，3 路输出有效。可写出其余 3 个输出表达式如下：

$$\overline{Y}_1 = \overline{\overline{A}_1A_0} = \overline{m}_1,\quad \overline{Y}_2 = \overline{A_1\overline{A}_0} = \overline{m}_2,\quad \overline{Y}_3 = \overline{A_1A_0} = \overline{m}_3$$

基于上面的分析，可总结 $\overline{Y}_0 \sim \overline{Y}_3$ 的通用输出表达式为

$$\overline{Y}_i = \overline{m}_i \tag{4.3.2}$$

2-4 译码器参考电路如图 4.3.17 所示。图中，$\overline{S}_T$ 为选通控制端，当 $\overline{S}_T=1$ 时，禁止译码(输出全“1”)。当 $\overline{S}_T=0$ 时，译码器工作，可写出如式(4.3.1)所示的各输出函数表达式。二进制译码器又称为变量译码器，各输出对应输入变量的全部最小项。

将图 4.3.17 输入到 Quartus Ⅱ中编译并仿真，得到的波形如图 4.3.18 所示。图中，A_1、A_0 为译码器的输入端，Y_3、Y_2、Y_1、Y_0 为译码器的输出端(实际电路中，Y_3 等输出有上画线，表示低电平为有效输出，仿真软件不支持上画线等文本格式，请注意这两种标注的区别)。由图 4.3.18 可以写出如表 4.3.5 所示的 2 线-4 线译码器的真值表。如表中第 6 行，输入 A_1A_0 为 11，经译码器译码后第 4 个输出 Y_3、A_0 有效(别忘了编码时为从 0 开始)，可

类似分析其他各行。基于译码器的每个编码均有输出与之对应，常形象地把输入 A_1、A_0 称为地址输入端。

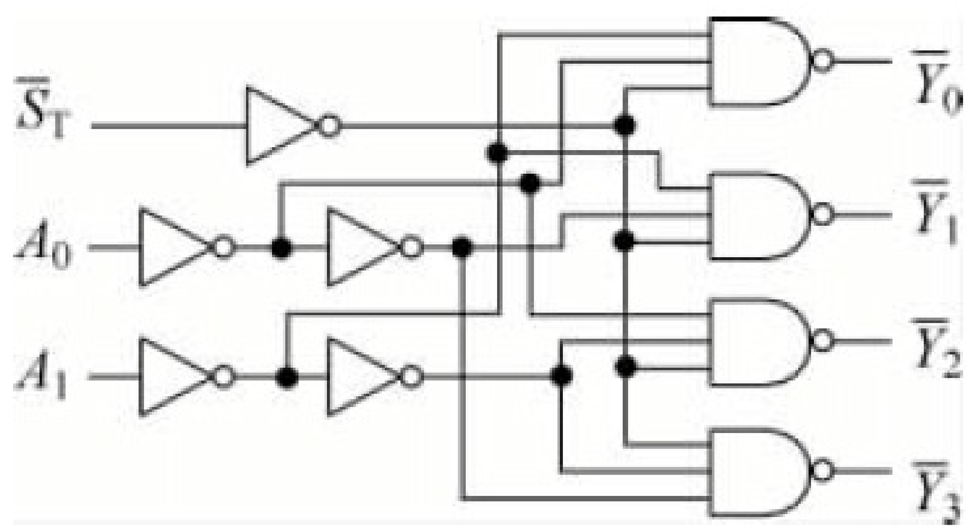


图 4.3.17　2-4 译码器

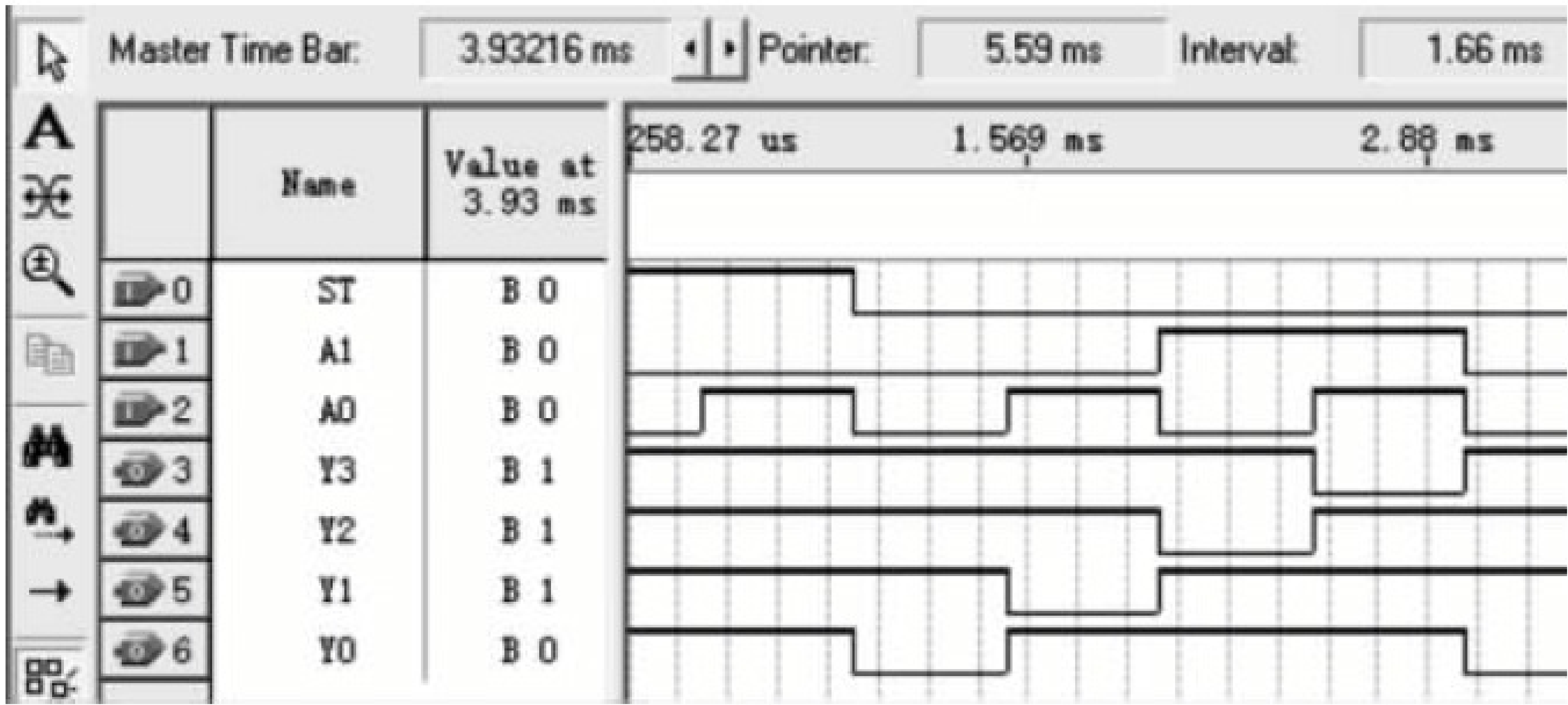


图 4.3.18　2-4 译码器仿真图

表 4.3.5　2-4 译码器真值表

$\bar{S}_T$	A_1	A_0	$\bar{Y}_0$	$\bar{Y}_1$	$\bar{Y}_2$	$\bar{Y}_3$
1	×	×	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

从仿真波形、真值表可看出，如果规定 0 为有效输出，则译码器输出端哪一条有输出信号，取决于输入的地址信号。

译码器的译码功能应用十分广泛，常用于多设备的协同工作。例如，计算机中具有音箱、打印机、显示器等各种设备，这些设备均与计算机的数据线连接。当计算机给打印机送数据时，音箱、显示器等设备不能接收这些数据，否则音箱会乱响，显示为乱码。可利用译码器控制这些设备的使能端，在给打印机送数据前，给出打印机对应的地址信号，使打印机使能端有效，此时，在译码器的控制下，其余各设备的使能端均无效，确保各设备协同工作。

基于译码器译码逻辑功能特点，可类似总结 3-8 译码器、4-16 译码器等二进制译码器通用输出表达式为

$$\bar{Y}_i = \bar{m}_i$$

二—十进制译码器和二进制译码器并无本质区别，通用输出表达式也为

$$\bar{Y}_i = \bar{m}_i$$

当然，二—十进制译码器只有 10 个输出，因此，大于 9 的输入代码为无效输入。

3. 常用集成译码器的逻辑功能

译码器是常用组合逻辑芯片，应用十分广泛，相应的集成译码器产品也较多，按照输入、输出线的多少有 2-4 译码器、3-8 译码器、4-16 译码器、4-10 译码器等。



如图 4.3.19 所示为应用十分广泛的 3-8 译码器 74LS138。图中，A_2、A_1、A_0 为译码器的地址端；$\bar{Y}_0 \sim \bar{Y}_7$ 为译码器的输出端。S_T、$\bar{S}_1$、$\bar{S}_2$ 为控制端。当 $S_T=1$、$\bar{S}_1=\bar{S}_2=0$ 时，译码器工作，其输出函数 $\bar{Y}_0 \sim \bar{Y}_7$ 的表达式为

$$\bar{Y}_i = \bar{m}_i$$

如 $\bar{Y}_0=\bar{m}_0=\overline{\bar{A}\bar{B}\bar{C}}$，则可类似写出其他输出表达式。其真值表如表 4.3.6 所示。

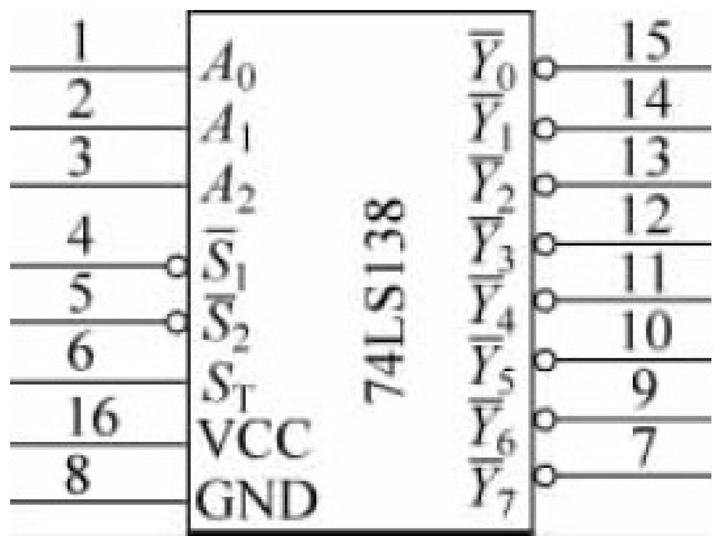


图 4.3.19 74LS138

表 4.3.6 74LS138 译码器真值表

S_T	$\bar{S}_1+\bar{S}_2$	A_2 A_1 A_0	输 出
0	×	× × ×	全 1
×	1	× × ×	全 1
1	0	0 0 0	$\bar{Y}_0=0$，其余为 1
1	0	m_i	$\bar{Y}_i=\bar{m}_i$，其余为 1

由表 4.3.6 可看出，$\bar{Y}_0 \sim \bar{Y}_7$ 为 A_2、A_1、A_0这 3 个变量的全部最小项的译码输出，所以也将这种译码器称为最小项译码器。

QuartusⅡ中库元件 74138 原理图如图 4.3.20 所示。其中的 G1、G2AN、G2BN 分别对应 3 个控制引脚 S_T、$\bar{S}_1$、$\bar{S}_2$，可通过该原理图理解 3 个控制引脚 S_T、$\bar{S}_1$、$\bar{S}_2$ 实现控制的方法，也可由该原理图直接写出输出 YON$=\bar{m}_0=\overline{\bar{A}\bar{B}\bar{C}}$(3 个控制引脚均有效时)，有兴趣的读者可进一步分析其他各输出的表达式。



可通过下面的例题理解 74LS138 设计 3 个控制引脚的好处。

【例 4.3.3】 试用两片 74LS138 接成 4-16 译码器。

解 图 4.3.21 所示为用两片 74LS138 级联起来构成的 4-16 译码器。分析如下。

(1) 输出设计。设片 1 为低 8 位，片 2 为高 8 位，分别对应 4-16 译码器的 16 个输出。

(2) 低 3 位 A_2、A_1、A_0。输入允许直接短接，可将片 1、片 2 的 A_2、A_1、A_0接在一起。

(3) 最高位地址 A_3。当 $A_3=0$ 时，片 1 工作，片 2 禁止工作。可令 A_3接片 1 的 $\bar{S}_1$ 及片 2 的 S_T。该接法满足当 $A_3=1$ 时片 2 工作，片 1 禁止工作的应用要求。

(4) 使能端的设计。片 1 的 $\bar{S}_2$，片 2 的 $\bar{S}_1$、$\bar{S}_2$ 未使用，将这 3 个控制引脚短接可形成整体的使能端 $\bar{S}$。当 $\bar{S}=1$ 时，级联电路被禁止，输出为全 1；当 $\bar{S}=0$ 时，级联电路工作。

由例 4.3.3 可看出，74LS138 具有 3 个控制引脚，非常方便级联。未用的控制引脚为恒有效，有兴趣的读者也可改进该电路，实现具有高、低两种电平控制方式的 4-16 译码器。

类似地，可用 3 片 74LS138 译码器级联构成 5-24 译码器；用 4 片 74LS138 译码器可以级联成 5-32 译码器。读者可参看有关资料。

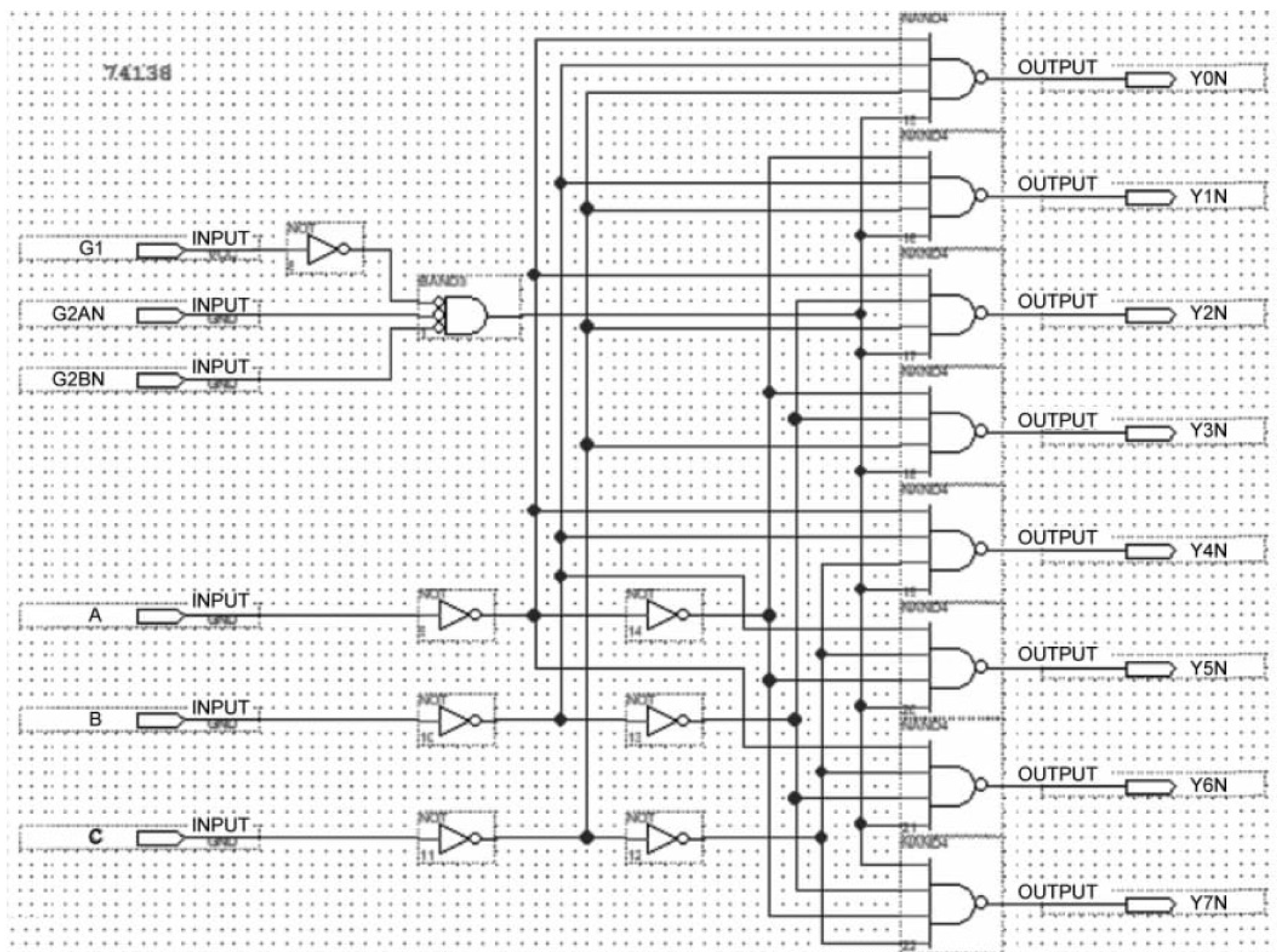


图 4.3.20　QuartusⅡ中库元件 74138 原理图

图 4.3.22 所示为应用十分广泛的 4-10 译码器 74LS42。图中 A_3、A_2、A_1、A_0为译码器的地址端；$\overline{Y}_0 \sim \overline{Y}_9$ 为译码器的输出端。它与前面介绍的二进制译码器基本类似，以低电平作为有效输出，其输出函数 $\overline{Y}_0 \sim \overline{Y}_9$ 的表达式也可用式(4.4.3)表示。不同的是，74LS42 只是将前 10 种编码(0～9)译码，对 10～15 的编码，译码输出全 1。

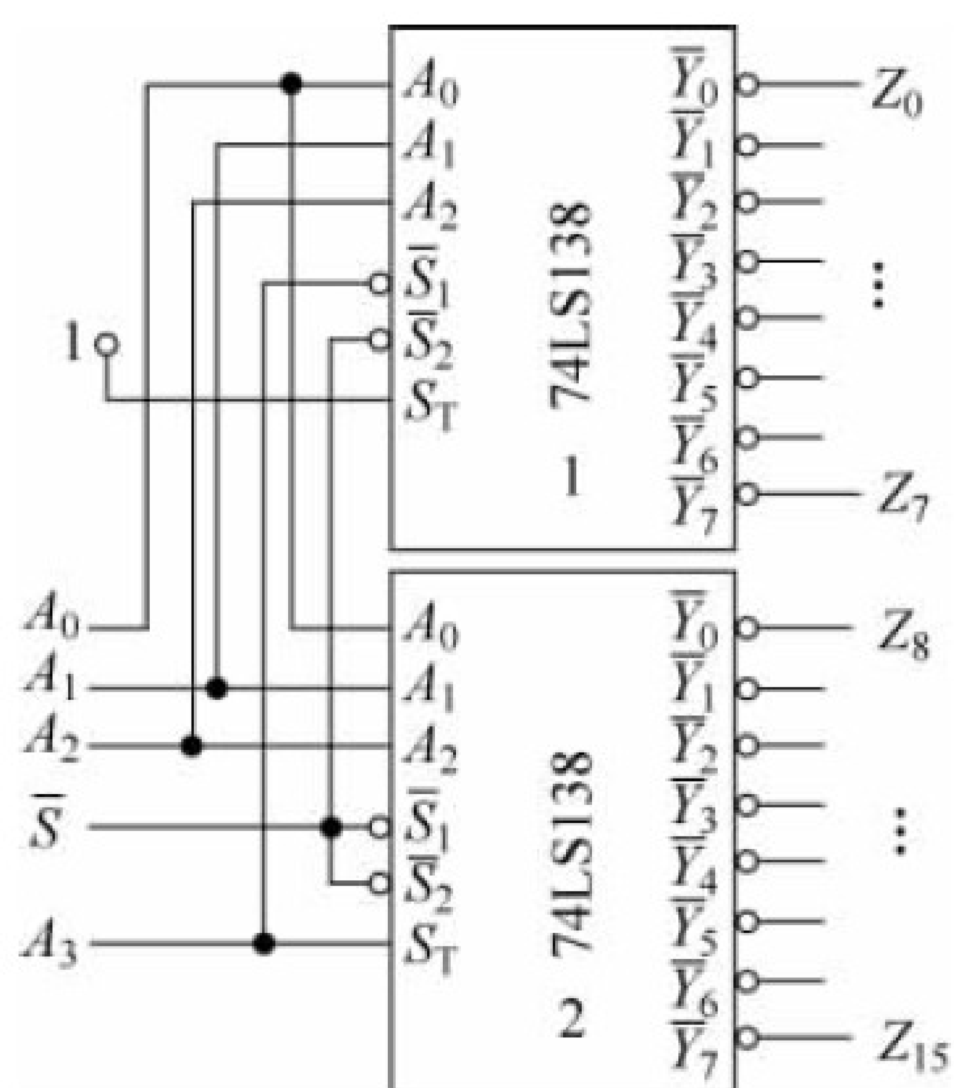


图 4.3.21　例 4.3.3 的电路图

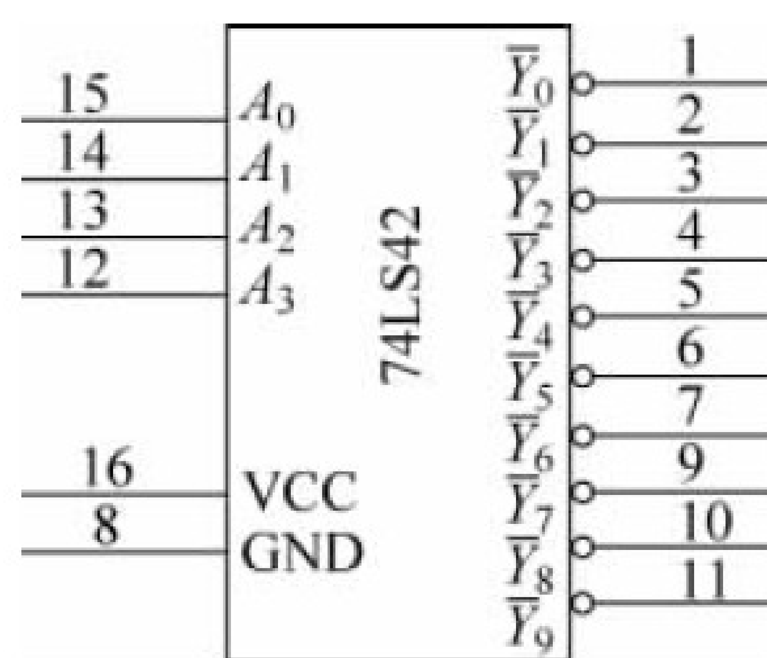


图 4.3.22　74LS42

QuartusⅡ中库元件 7442 原理图如图 4.3.23 所示。其中的 A、B、C、D 对应 4 个地址输入端 $A_3A_2A_1A_0$，O0N～O9H 分别对应 10 个输出端$\overline{v}_0 \sim \overline{v}_9$。由该原理图可直接写出输

出 O0N$=\bar{m}_0=\overline{\bar{A}\bar{B}\bar{C}\bar{D}}$，有兴趣的读者可进一步分析其他各输出的表达式。4-10 译码器 74LS42 的真值表如表 4.3.7 所示。

图 4.3.23 QuartusⅡ中库元件 7442 原理图

表 4.3.7 74LS42 的真值表

A_3 A_2 A_1 A_0	输 出
0 0 0 0	$\bar{Y}_0=0$，其余为 1
$m_i(i<10)$	$\bar{Y}_i=\bar{m}_i$，其余为 1

4. 数字显示译码器



数字显示译码器能将输入的数码翻译成数码管、液晶等数字显示设备需要的代码。显然，显示器件和显示方式不同，译码电路也不相同。因此，理解数字显示译码器，首先应理解数码管、液晶等数字显示设备的基本原理。

(1) 半导体数码管。用某些特殊的半导体材料，例如，用磷砷化镓做成的 PN 结，当外

加正向电压时,可以将电能转换成光能,从而发出清晰悦目的光线。利用这样的 PN 结,既可以封装成单个的发光二极管(Light Emitting Diode,LED),也可以分段式封装成数码管,其引脚排列如图 4.3.24 所示。中间两个引脚为 8 个 LED 的公共端。由于二极管具有单向导电性,因此,数码管具有共阴、共阳两种类型。图 4.3.25(a)中各个二极管的阳极相互连接组成公共端,为共阳数码管;图 4.3.25(b)中各个二极管的阴极相互连接组成公共端,为共阴数码管。

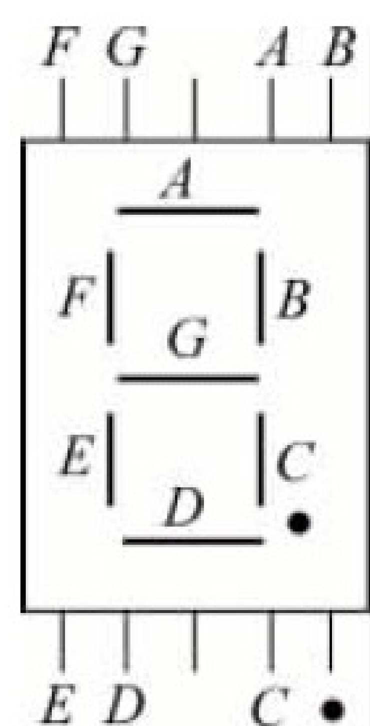


图 4.3.24 数码管的引脚排列

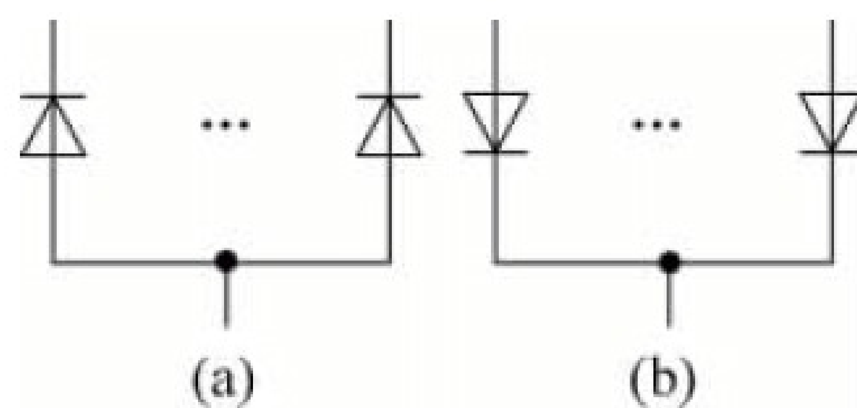


图 4.3.25 共阳、共阴模型

对共阴数码管,若将公共端接地,$A \sim G$ 共 7 个引脚接高电平,则七段 LED 全亮,显示为 8。对共阳数码管,若将公共端接高电平,$A \sim G$ 共 7 个引脚接低电平,则数码管显示也为 8。

半导体数码管具有工作电压低、体积小、寿命长、可靠性高、响应时间短(一般不超过 0.1μs)、亮度高等诸多优点;不足为工作电流较大,一般约为 10mA。

(2) 液晶七段字符显示器。另一种常用的七段字符显示器是液晶显示器(Liquid Crystal Display,LCD)。液晶是介于液态与结晶态之间的一种物质状态,兼有液体和晶体的某些性质。液晶材料主要是脂肪族、芳香族、硬脂酸等有机物。显示原理如下:

将液晶置于两片导电电极之间,外盖两片玻璃盖板。正常状态下,没有外加电场,液晶分子按一定取向整齐排列,如图 4.3.26(a)所示。这时,液晶呈透明状态,射入的光线大部分由反射电极反射回来,显示器呈白色。当显示器外加电场时,液晶分子将因为电离产生正离子,这些正离子将在电场作用下运动并碰撞其他液晶分子,使液晶分子无法保持整齐排列,液晶呈混沌状态,如图 4.3.26(b)所示。这时,射入的光线散射后只有少量反射回来,显示器呈暗灰色。当外加电场消失后,液晶将重新恢复到整齐排列状态。类似地,将透明玻璃盖板换成彩色滤光片,则可显示彩色影像。

可见,液晶的透明度和呈现的颜色受外加电场的影响。利用这个特点,将七段透明的电极排列成 8 字,可将液晶制作成七段字符显示器。

液晶显示材料主要优点:驱动电压低、功耗微小(小于 $1\mu W/cm^2$)、可靠性高、显示信息量大、彩色显示无闪烁、对人体无危害、便于携带,可以制成各种规格和类型的液晶显示器等。

由于这些优点,用液晶材料制成的计算机终端和电视可以大幅度减小体积,液晶显示

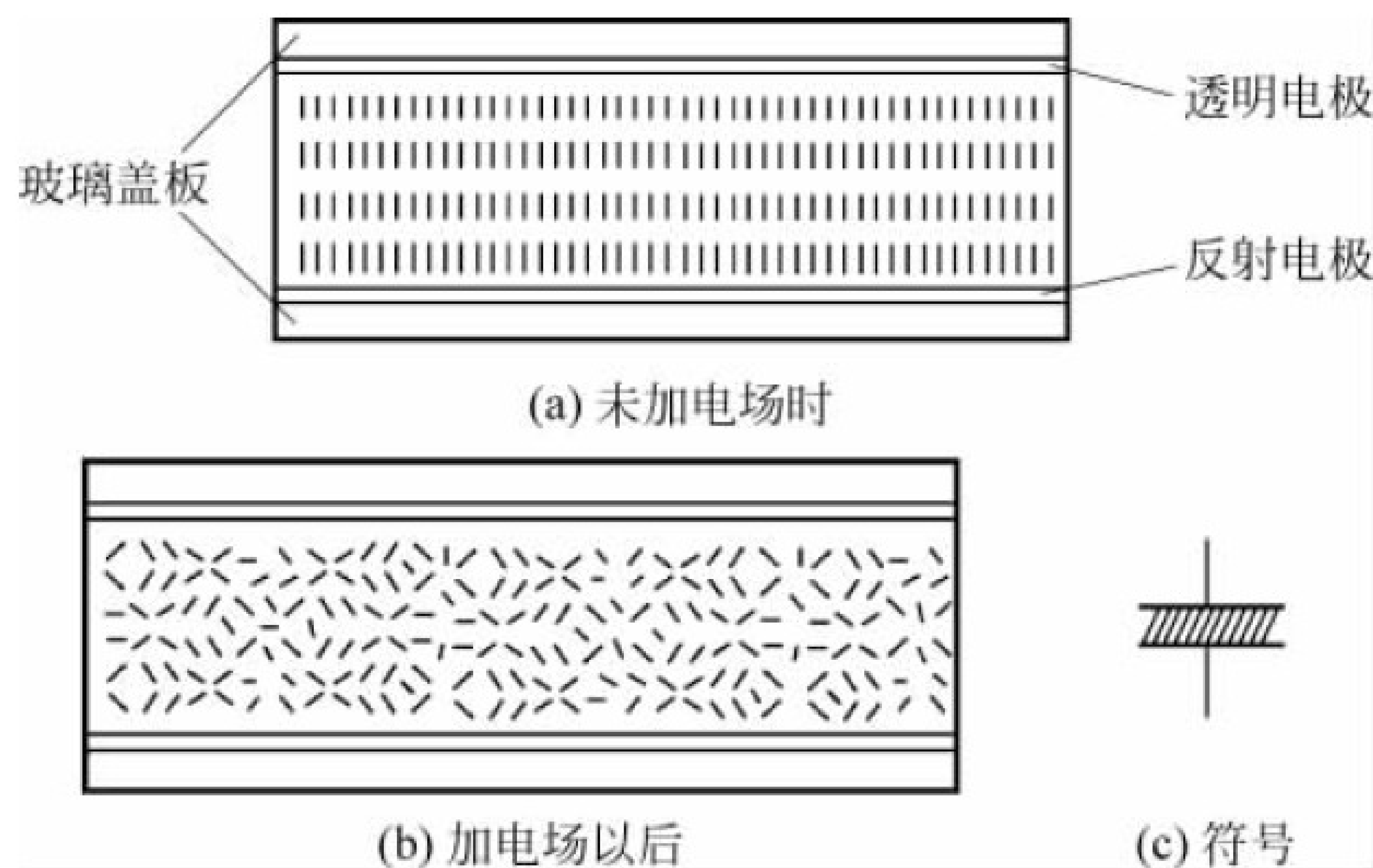


图 4.3.26 液晶显示器的结构及符号

技术对显示显像产品结构产生了深刻影响，促进了微电子技术和光电信息技术的发展。

必须指出的是，液晶自身不会发光，必须靠反射外界光才能显示具体颜色，其显示原理、驱动方式比 LED 显示器复杂得多。此外，与 LED 显示器相比，液晶显示器亮度差，响应速度较低(10～200ms)。

本书主要介绍半导体数码管的显示及其驱动方法，关于液晶显示器的显示原理、驱动方式方法等方面的更多知识，请读者参考相关书籍。

(3) BCD-七段显示译码器。BCD-七段显示译码器的功能是将 8421BCD 代码译成对应的数码管的 7 个字段信号，驱动数码管，显示出相应的十进制数码，原理框图如图 4.3.27(a)所示。由图可看出，BCD-七段显示译码器输入为 4 位二进制 BCD 码，输出为 7 个字段驱动电平。显然，七段显示译码器每个输入代码对应的输出不是某一根输出线上的高、低电平，而是另一个 7 位的代码，明显不同于前面介绍的译码器。严格地讲，将这种电路称为代码变换器更确切些。但习惯上都把它称为显示译码器。

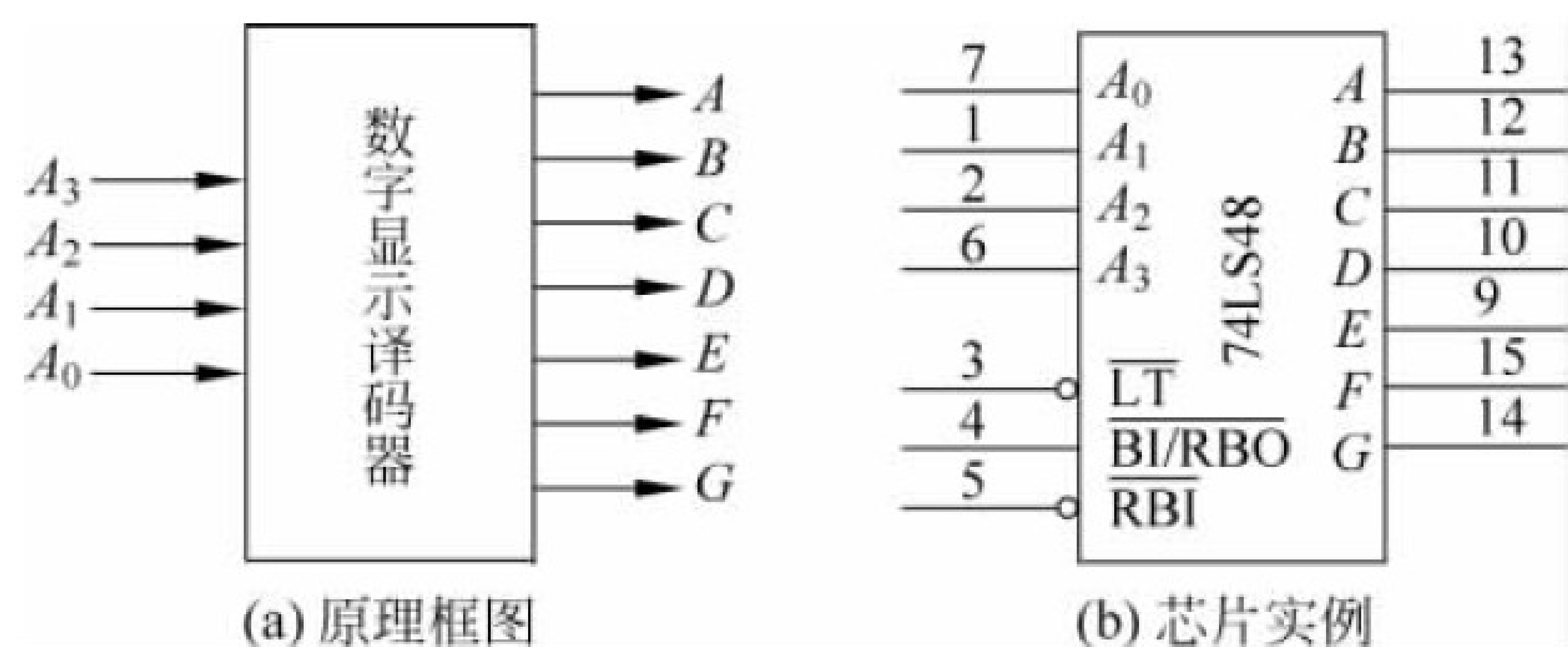




图 4.3.27 七段显示译码器

七段显示译码驱动芯片种类较多，驱动共阳数码管的译码芯片有 74LS47、74LS247 等；驱动共阴数码管的译码芯片有 74LS48、74LS248 等。

如图 4.3.27(b)所示为驱动共阴数码管的译码芯片 74LS48 引脚图。其中，$A_3 \sim A_0$为 4 位二进制输入码，$A \sim G$ 为 7 个字段输出信号，$\overline{BI/RBO}$(复用引脚，当分别作为输入和输出引脚时，具有不同的功能)、$\overline{RBI}$、$\overline{LT}$为 3 个控制引脚。

74LS48 功能表如表 4.3.8 所示。真值表第 1 行表明，当$\overline{\mathrm{LT}}=0$且$\overline{\mathrm{BI}}$输入引脚无效时，$A\sim G$共 7 个字段输出全为 1，7 个 LED 均被点亮。此时，如果数码管实际显示 8，说明数码管的 7 个 LED 均正常，$\overline{\mathrm{LT}}$称为灯好坏测试输入引脚。真值表第 2 行表明，$\overline{\mathrm{BI}}$为灭灯输入引脚，当$\overline{\mathrm{BI}}=0$时，$A\sim G$共 7 个字段输出全为 0，7 个 LED 均不亮。真值表第 3 行表明，$\overline{\mathrm{RBI}}$为灭零输入引脚。当$\overline{\mathrm{RBI}}=0$且$\overline{\mathrm{BI}}$、$\overline{\mathrm{LT}}$输入引脚无效时，数码管数码 0 不显示，其余数码正常显示。

表 4.3.8　74LS48 功能表

输入输出 / 说明	$\overline{\mathrm{LT}}$	$\overline{\mathrm{RBI}}$	$\overline{\mathrm{BI}}$	$A_3\ A_2\ A_1\ A_0$	$A\ B\ C\ D\ E\ F\ G$	显　示
测试	0	×	1	× × × ×	1 1 1 1 1 1 1	8
灭灯	×	×	0	× × × ×	0 0 0 0 0 0 0	不亮
灭 0	1	0	1	0 0 0 0	0 0 0 0 0 0 0	灭 0
显示 0	1	1	1	0 0 0 0	1 1 1 1 1 1 0	0
正常译码	1	×	1	0 0 0 1 …	0 1 1 0 0 0 0 …	1 …

由表 4.3.8 第 5、6 行表明，74LS48 用于正常译码时，$\overline{\mathrm{BI}}=\overline{\mathrm{RBI}}=\overline{\mathrm{LT}}=1$。当$\overline{\mathrm{BI}}$、$\overline{\mathrm{RBI}}$、$\overline{\mathrm{LT}}$接电源、7 个输出接数码管的对应引脚(图 4.3.28)，输入 $A_3A_2A_1A_0=1$ 时，74LS48 将产生让共阴数码管显示数字 1 的七段字型码“0110000”，即让数码管的 B、C 段发光，类似输入 $A_3A_2A_1A_0=2$ 时数码管显示数字 2。

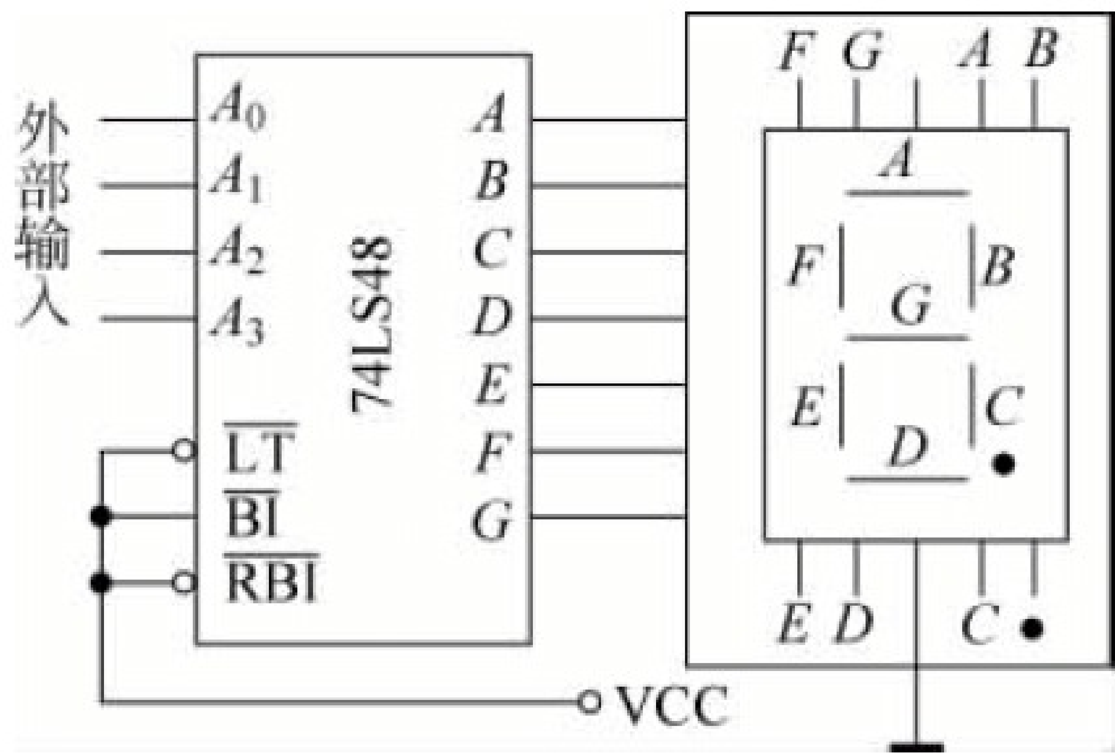


图 4.3.28　74LS48 与数码管的连接

前面指出，$\overline{\mathrm{BI}/\mathrm{RBO}}$为复用引脚，可作为输出引脚使用。当输入 $A_3\sim A_0$为“0000”，且$\overline{\mathrm{LT}}$无效、$\overline{\mathrm{RBI}}$有效时，该引脚输出为低电平，可作为灭零输出引脚使用。

灭零应用方法如图 4.3.29 所示。图中，左边数码管的驱动控制$\overline{\mathrm{LT}}=1$，$\overline{\mathrm{RBI}}=0$。当输入 $A_3\sim A_0$为“0000”时，该数码管的数码 0 不显示，$\overline{\mathrm{RBO}}$输出低电平。$\overline{\mathrm{RBO}}$与右边数码管的驱动控制灭零输入$\overline{\mathrm{RBI}}$连接，$\overline{\mathrm{RBI}}=0$。若此时该数码管的输入 $A_3\sim A_0$也为“0000”，则该数码管的数码 0 也将不显示。这种因为高位数码管的“0”不显示将使低位数码管的零也不显示的控制方法便是典型的灭零控制应用方法。

参考上面的灭零控制原理，3 位整数的参考显示应用系统如图 4.3.30 所示。图中，个数位的数码全部显示，百位数的灭零输入端恒有效，百位数的 0 将直接不显示。百位数的灭零输出端接十位数的灭零输入端，当百位、十位上的数码均为 0 时，十位上的数码 0 也将不显示。

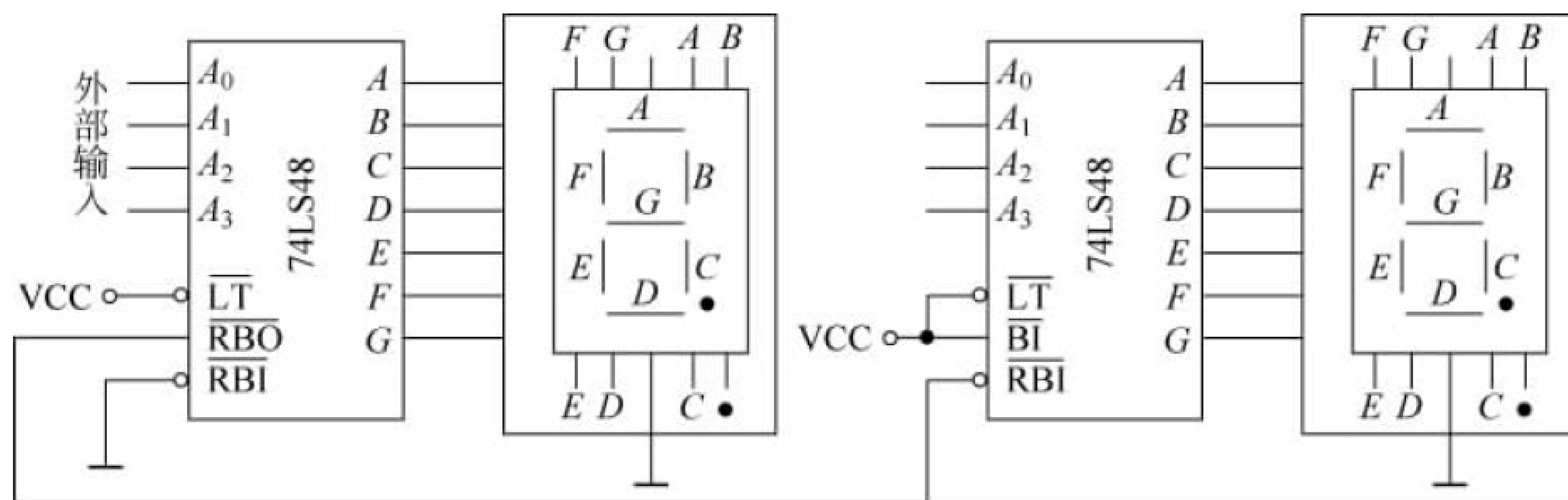


图 4.3.29　74LS48 灭零应用方法

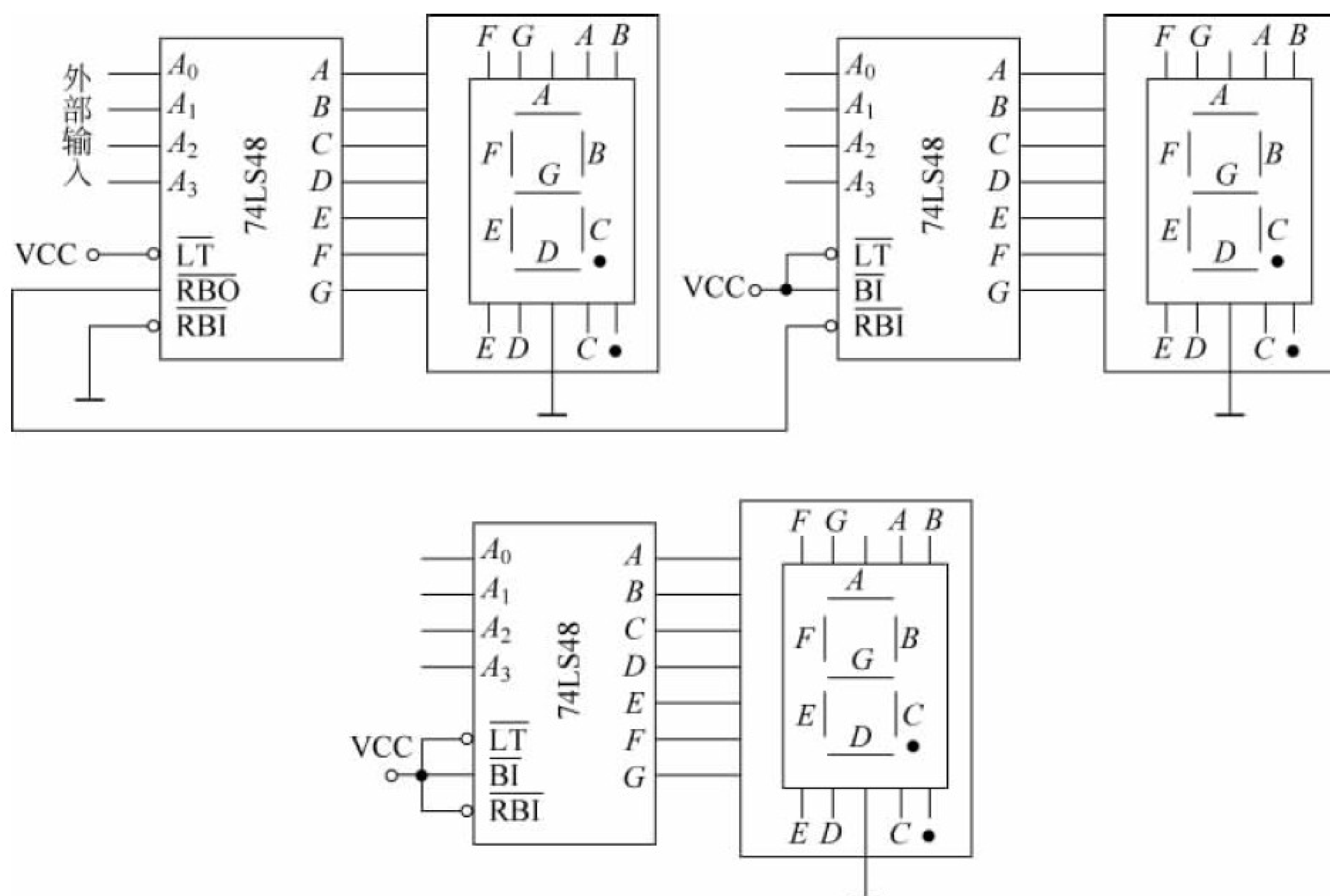


图 4.3.30　典型灭零应用系统

当然，实际的数据显示系统中可能涉及小数，当小数点位置固定时，可将该位置上数码管的小数点对应的 LED 控制端固定接高电平(共阴数码管)，利用 74LS48 驱动各数码管并根据灭零要求设置各灭零输入控制，有兴趣的读者请参考相关书籍。

QuartusⅡ中库元件 7448 原理图如图 4.3.31 所示。图中，A、B、C、D 对应 $A_3A_2A_1A_0$，OA～OG 分别对应 7 个字段输出端 A～G。LTN、RBIN 对应控制引脚$\overline{\mathrm{LT}}$、$\overline{\mathrm{RBI}}$。输入引脚 BIN、输出引脚 RBON 组合构成复用引脚$\overline{\mathrm{BI}/\mathrm{RBO}}$。

由该原理电路，可写出灭零输出为

$$\mathrm{RBO}=\overline{\mathrm{LTN}\ \overline{\mathrm{RBIN}}\ \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}} \tag{4.3.3}$$

参考式(4.3.3)，可写出 74LS48 的灭零输出函数为

$$\overline{\mathrm{RBO}}=\overline{\overline{\mathrm{LT}}\cdot\mathrm{RBI}\cdot\overline{A}_3\cdot\overline{A}_2\cdot\overline{A}_1\cdot\overline{A}_0} \tag{4.3.4}$$

可见，当 74LS48 的$\overline{\mathrm{LT}}=1$、$\overline{\mathrm{RBI}}=0$ 且输入 A_3～A_0 为“0000”时，$\overline{\mathrm{RBO}}$输出低电平。根

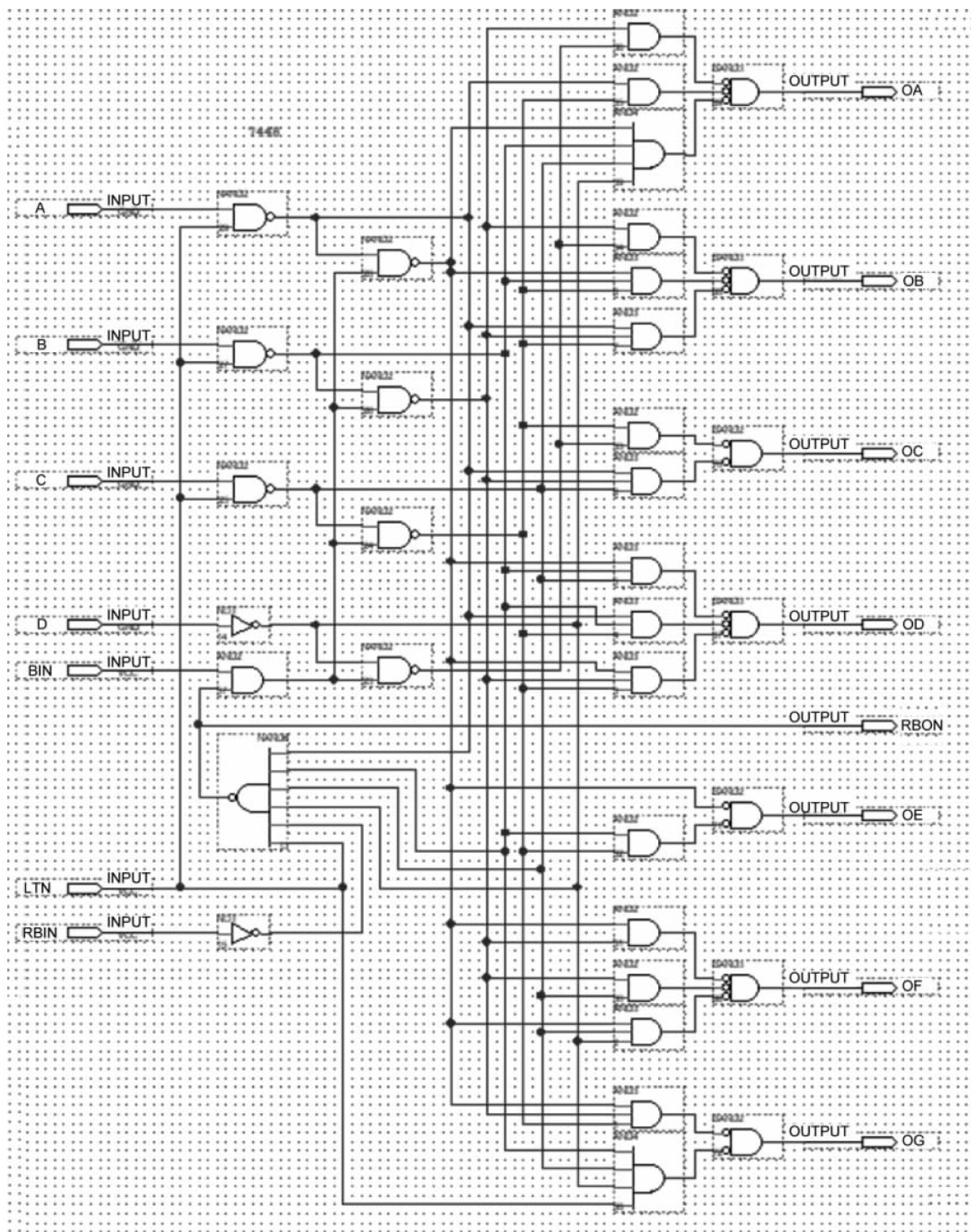


图 4.3.31 QuartusⅡ中库元件 7448 原理图

据该输出函数，74LS48 的灭零输出不会影响自身的灭零输入的状态。有兴趣的读者可仔细分析该电路并进一步理解 74LS48 的显示译码功能及其控制引脚功能的实现。

由于 74LS48 等芯片无记忆功能，所以当外部输入消失后，数码管将不再显示过去的数据，为此，在实际显示系统中必须以某种方式使数码管一直显示需要的数据。解决的方法

之一是使用带锁存的芯片(如4511),有兴趣的读者请查阅相关资料。

4.3.3 加法器

计算机中,减法通过加法实现,加法器是构成算术运算器的基本单元。



1. 半加器

不考虑来自低位的进位的两个1位二进制数的加法运算,称为半加运算。实现半加运算的电路称为半加器,其电路如图4.3.32所示。将如图4.3.31所示电路输入到Quartus Ⅱ中编译并仿真,得到的波形如图4.3.33所示。分别令A、B为两个1位二进制数的加数、被加数,S为相加后的和数,CO为向高一位的进位数,则由图4.3.33可知如图4.3.32所示电路为半加器电路。如图4.3.34所示为半加器的逻辑符号。

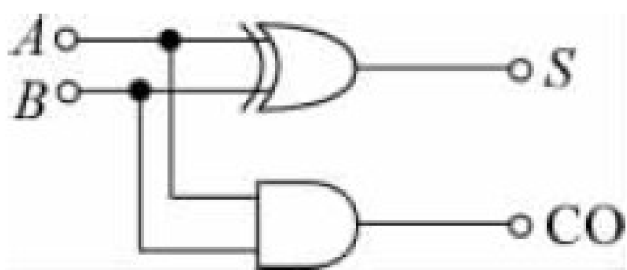


图4.3.32 半加电路

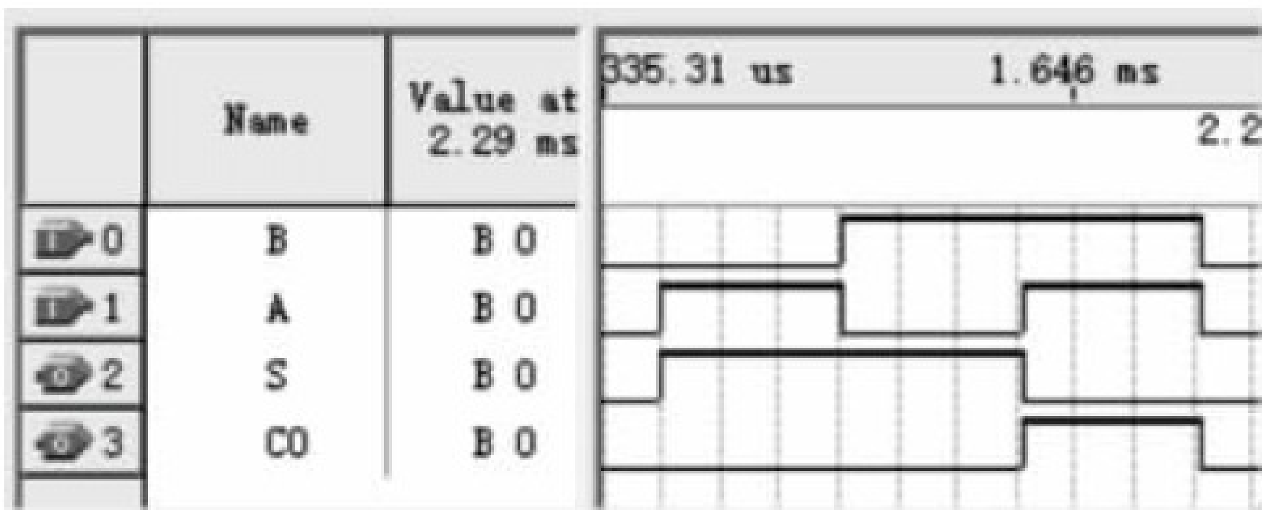


图4.3.33 半加器仿真图

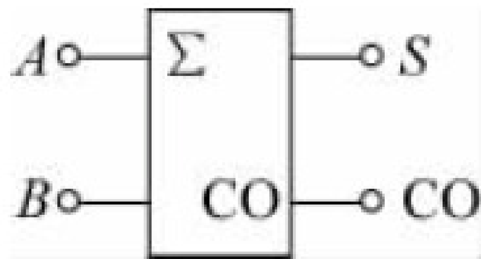


图4.3.34 半加器逻辑符号

也可通过门电路的特点直接理解半加器电路的逻辑功能。S为A、B的异或。前面指出,异或为不一样的"或",为真正的加。可见,S为A、B的和。CO为A、B的与,符合两个1位二进制数的加法的进位特点,为向高一位的进位。

2. 全加器

考虑来自低位进位数的两个1位二进制数的加法运算,称为全加运算。实现全加运算的电路称为全加器,其电路如图4.3.35所示。将如图4.3.35所示的电路输入到Quartus Ⅱ中编译并仿真,得到的波形如图4.3.36所示。分别令A、B为两个1位二进制数的加数、被加数,CI为来自低位的进位数;S为相加后的和数,CO为向高一位的进位数,则由图4.3.36可知如图4.3.35所示电路为全加器电路。如图4.3.37所示为全加器的逻辑符号。

也可以从另一个角度理解全加器:全加器由两个半加器加一个或门构成,其连接方法如图4.3.38所示。即用半加器1将A、B两个数相加,其和再与CI用半加器2相加所得的和为最终的和S;半加器1的进位与半加器2的进位之或构成向高一位的进位CO。

3. 串行进位加法器

由图4.3.38可知,全加器的加法过程为(忽略或门运算时间):半加器1相加完成后再

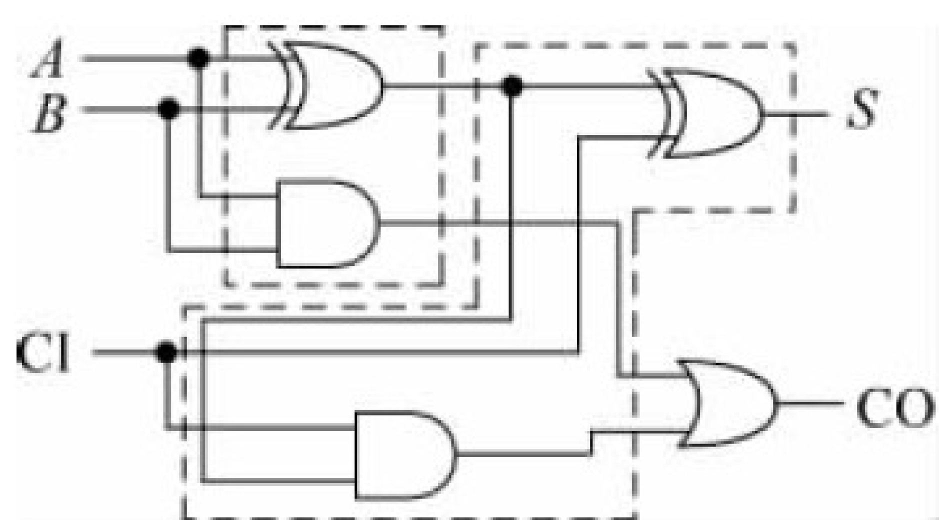


图 4.3.35　全加器电路

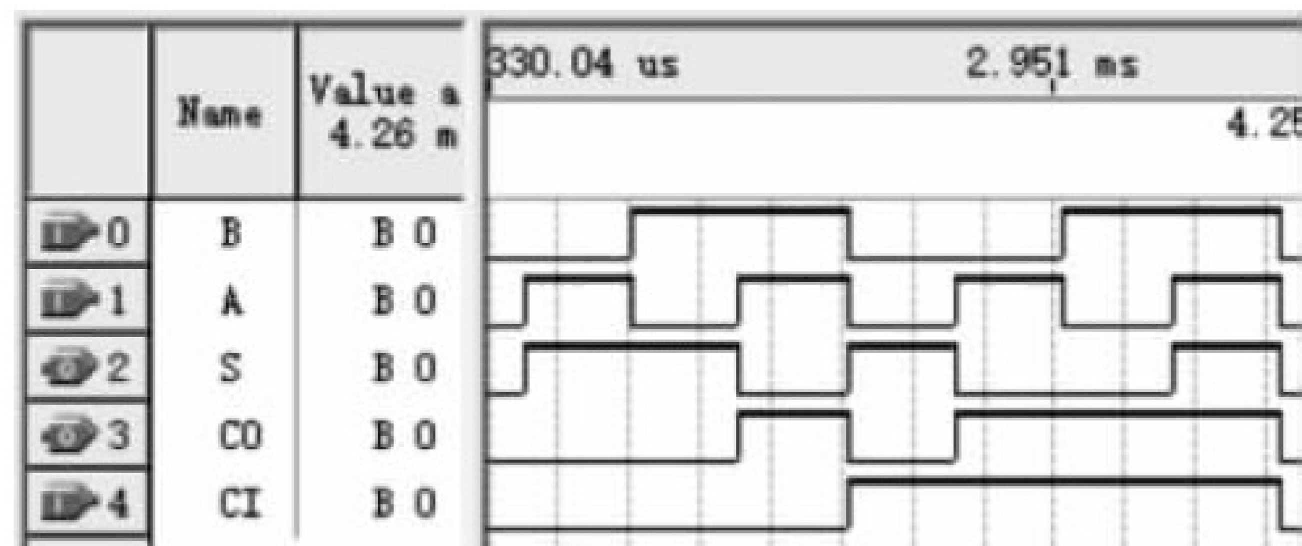


图 4.3.36　全加器仿真图

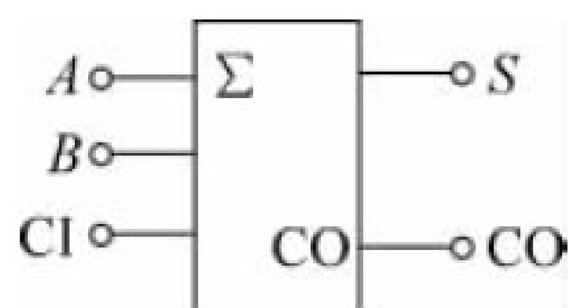


图 4.3.37　全加器逻辑符号

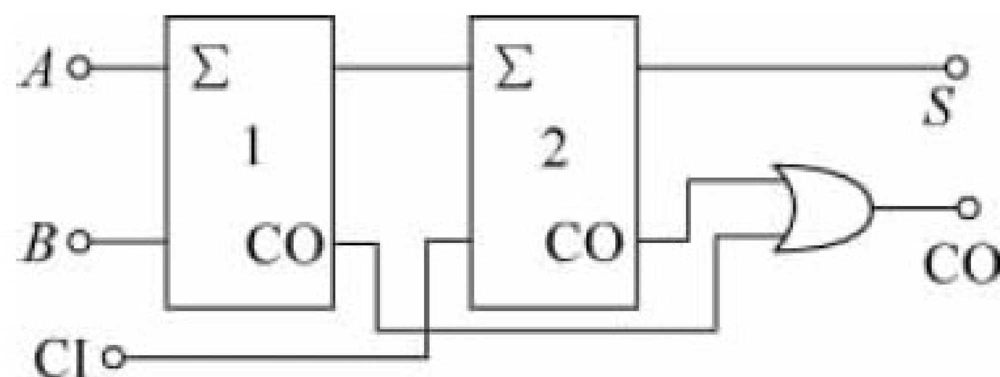


图 4.3.38　由半加器构成的全加器

由半加器 1 的结果及低位进位由半加器 2 相加后求出最终的和。全加器的最终进位由半加器 1、半加器 2 的进位相或决定。

这种加法器高位的运算需要等待低位运算所产生的进位才可求得，称它为串行进位加法器。可按照这种方法用全加器构成多位加法器。用全加器构成的两位串行进位加法器电路如图 4.3.39 所示。图中，S_0为 A_0、B_0的和，可直接求解。S_1需要等待 A_0、B_0相加后的进位才可求出最终结果，为典型的串行进位加法器。

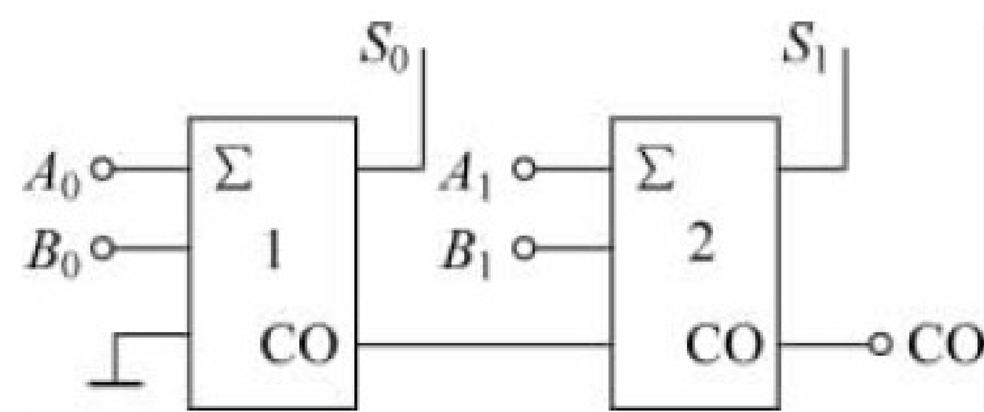


图 4.3.39　两位串行进位加法器

串行进位加法器电路结构比较简单。但是，这种电路的最大缺点是运算速度慢，仅在对运算速度要求不高的设备中采用。为提高运算速度，人们又设计了超前进位的加法器。

4. 超前进位加法器



所谓超前进位加法器，是指在做加法运算时，各位数的进位信号由输入的二进制数直接产生的加法器。

超前进位全加器超前进位原理如下：

$$\mathrm{CO}_i = A_iB_i + (A_i + B_i)\mathrm{CI}_i \tag{4.3.5}$$

式中，CO_i为向高位的进位输出；A_i、B_i分别为加数和被加数；CI_i为来自低位的进位，等于来自低位的进位输出，有

$$\mathrm{CI}_i = \mathrm{CO}_{i-1} \tag{4.3.6}$$

由式(4.3.5)和式(4.3.6)可知，在下面两种情况下，第 i 位向高位的进位输出为 1：

(1) 第 i 位的加数、被加数 A_i、B_i均为 1，$A_iB_i=1$。

(2) 第 i 位的加数、被加数至少有 1 个为 1，来自低位的进位输出为 1，$(A_i+B_i)\mathrm{CO}_{i-1}=1$。

可总结超前进位全加器超前进位的类推公式如下：

$$CO_i = A_iB_i + (A_i + B_i)CO_{i-1} \tag{4.3.7}$$

由式(4.3.7),有

$$CO_0 = A_0B_0 + (A_0 + B_0)$$

$$CO_1 = A_1B_1 + (A_1 + B_1)CO_0 = A_1B_1 + (A_1 + B_1)(A_0B_0 + (A_0 + B_0))$$

$$\vdots$$

显然,超前进位加法器运算速度的提高是以增加电路的复杂程度为代价来实现的。当加法器的位数增加时,电路的复杂程度将随之急剧增加。

如图 4.3.40 所示为超前进位的 4 位全加器 74LS283 的引脚图。图中,$A_4A_3A_2A_1$、$B_4B_3B_2B_1$ 分别为四位的加数及被加数,$S_4S_3S_2S_1$ 为和,CI 为来自低位的进位,CO 为向高位的进位。如输入 $A_4A_3A_2A_1$(1000)、$B_4B_3B_2B_1$(0110)、CI(0),则输出为 $S_4S_3S_2S_1$(1110)、CO(0)。

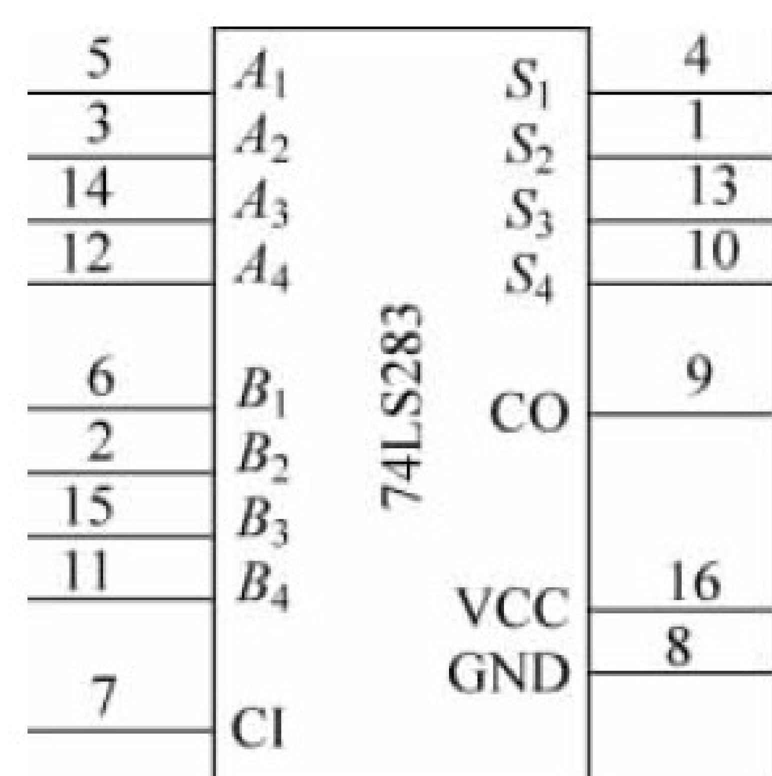


图 4.3.40　74LS283

74LS283 原理电路如图 4.3.41 所示。图中,第 2 位的和为

$$S_2 = ((A_2 + B_2)\overline{A_2B_2}) \oplus CO_1 = (A_2\,\overline{A_2B_2} + B_2\,\overline{A_2B_2}) \oplus CO_1$$

$$= (A_2\bar{B}_2 + B_2\bar{A}_2) \oplus CO_1 = (A_2 \oplus B_2) \oplus CO_1 = (A_2 \oplus B_2) \oplus CI_2$$

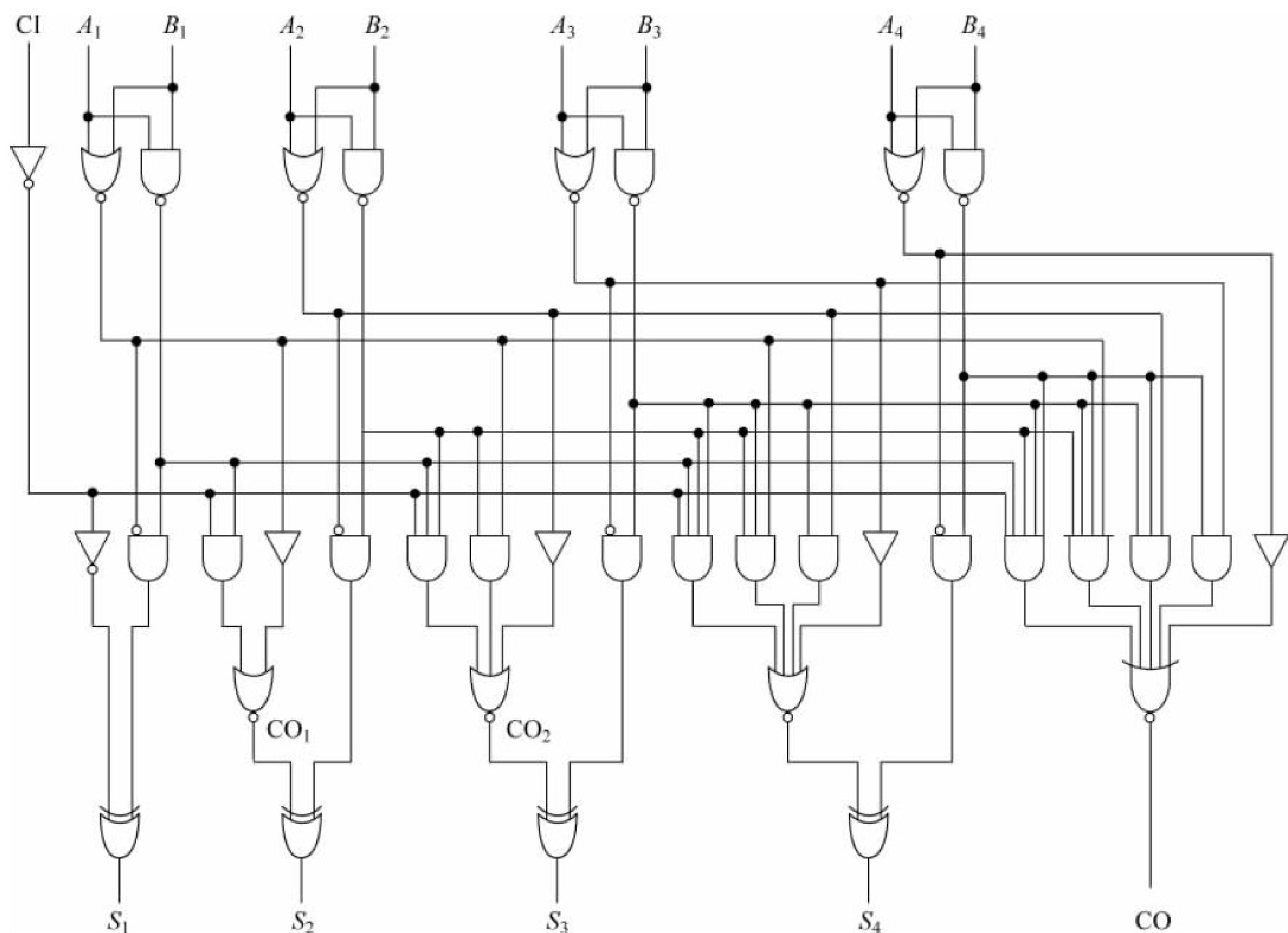


图 4.3.41　74LS283 原理电路

可见，S_2的逻辑表达式和如图 4.3.35 所示的全加器电路的求和电路功能相同，实现了 A_2、B_2及来自低位的进位的全加功能。

超前进位输出为

$$\mathrm{CO}_1 = \overline{\overline{A_1 + B_1} + \overline{\mathrm{CI}}\,\overline{A_1B_1}} = (A_1 + B_1)\,\overline{\overline{\mathrm{CI}}\,\overline{A_1B_1}}$$

$$= (A_1 + B_1)(\mathrm{CI} + A_1B_1) = (A_1 + B_1)\mathrm{CI} + A_1B_1$$

有兴趣的读者可参考上面的分析方法进一步分析其他各二进制位全加的实现方法。

加法运算是计算机中的基础运算，实践中，加法器应用十分广泛。如图 4.3.42 所示为 8421BCD 码转换为余 3 码的电路。依照编码规则，余 3 码的值比 8421BCD 码的值大 3，利用 74LS283 将输入的 4 位 8421BCD 码加上二进制数"0011"即可实现转换。

必须指出，在计算机中，减法是用加法实现的，因此，加法器不仅可以做加法，还可以实现减法运算。如图 4.3.43 所示为两个 4 位二进制正数的减法运算电路。其中，$C_3C_2C_1C_0$为被减数，$D_3D_2D_1D_0$为减数，$Y_3Y_2Y_1Y_0$为结果，BO 为借位输出。

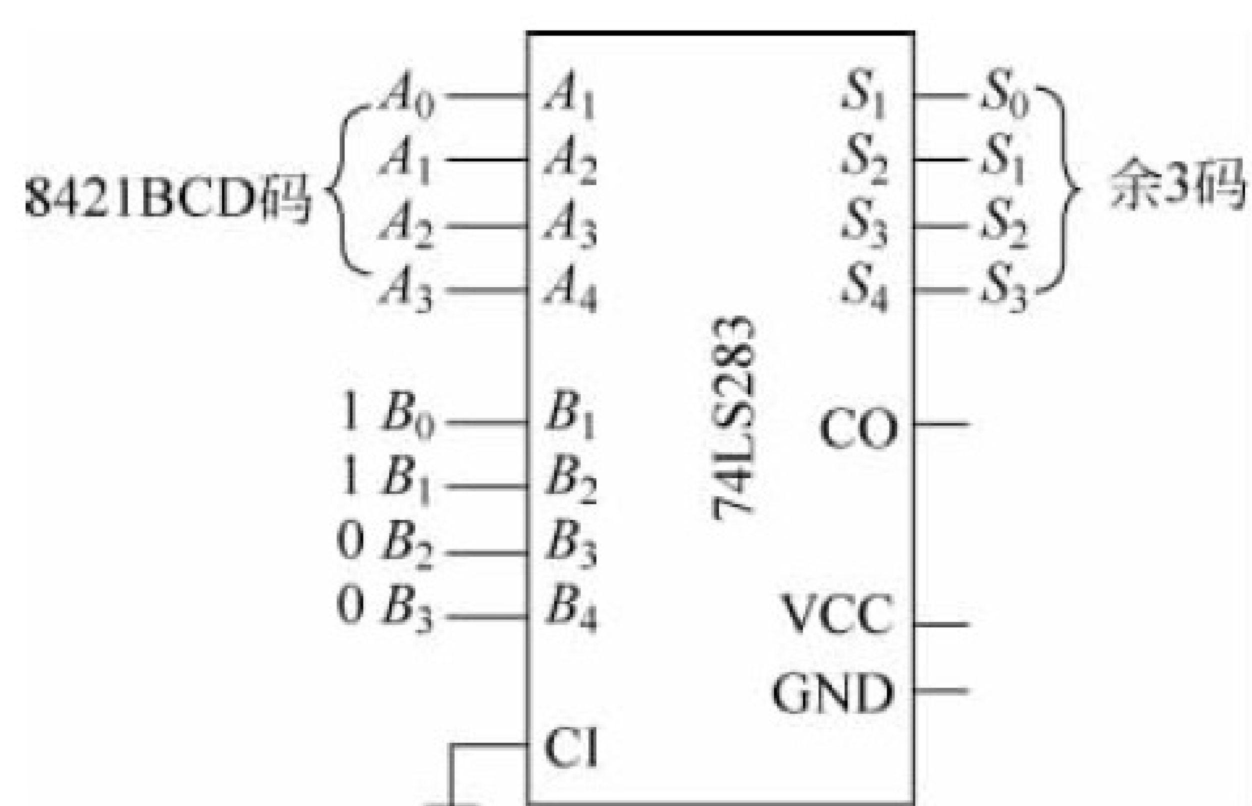


图 4.3.42　8421BCD 码转换为余 3 码电路

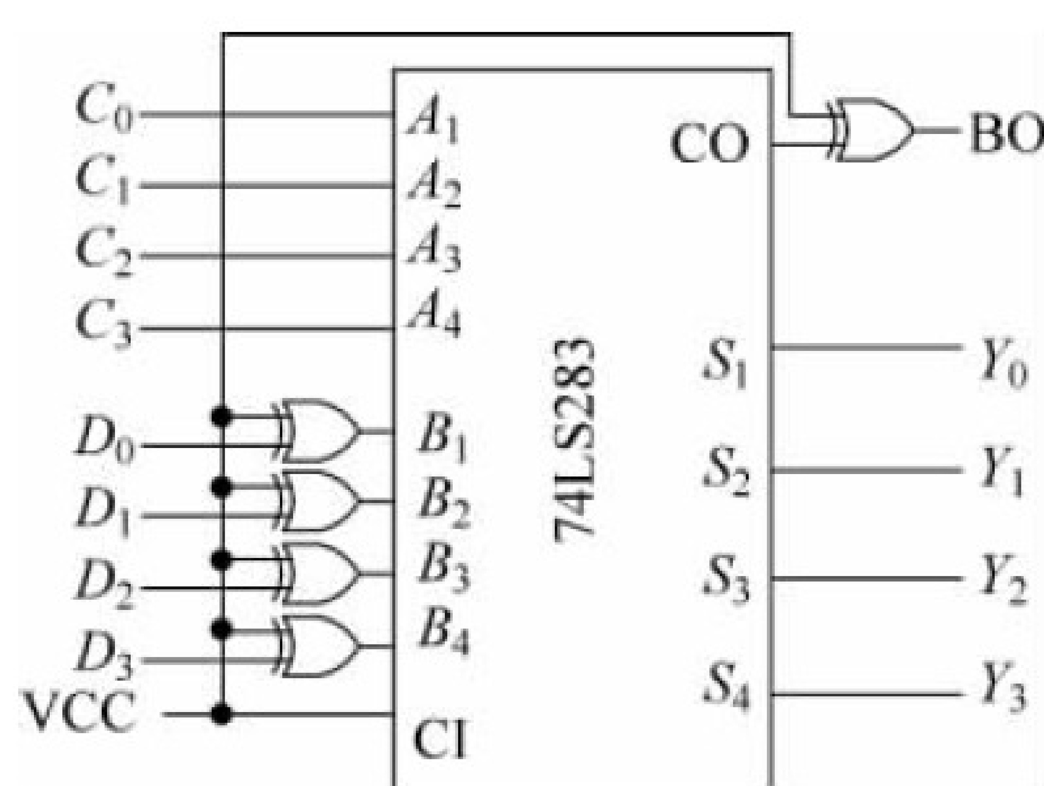


图 4.3.43　两个 4 位二进制正数的减法运算电路

图 4.3.43 中，两个 4 位二进制 $C_3C_2C_1C_0$、$D_3D_2D_1D_0$均为原码，减数 $D_3D_2D_1D_0$每位与高电平异或，来自低位的进位为高电平。减数每位与高电平异或，相当于逐位取反。加法器加上来自低位的进位，实现了对减数 $D_3D_2D_1D_0$的"逐位取反加 1"，求出了减数的补码。因此，如图 4.3.43 所示为两个 4 位二进制正数的减法运算电路。



4.3.4　数据选择器

在多路数据传送过程中，往往需要将多路数据中任意一路信号挑选出来，能实现这种逻辑功能的电路称为数据选择器（或者称为多路选择器、多路开关）。可通过图 4.3.44 来理解数据选择器。



图 4.3.44　数据选择器

图 4.3.44 中，D_0、D_1、D_2、D_3为 4 路输入信号，A_1、A_0为选择控制信号，Y 为输出信号。Y 可为 4 路输入数据中的任意一路，究竟是哪一路完全由地址选择控制信号 A_1、A_0决定。

按照逻辑功能要求，可令：$A_1A_0 = 00$ 时，$Y = D_0$；$A_1A_0 = 01$

时，$Y=D_1$；$A_1A_0=10$ 时，$Y=D_2$；$A_1A_0=11$ 时，$Y=D_3$。按照上述设计的逻辑电路可完成四选一的逻辑功能。上面的分析可写成如下的表达式：

$$Y = D_0\overline{A}_1\overline{A}_0 + D_1\overline{A}_1A_0 + D_2A_1\overline{A}_0 + D_3A_1A_0 = \sum_{i=0}^{3} D_i m_i \qquad (4.3.8)$$

数据选择器应用十分广泛，集成数据选择器的规格品种较多，如双四选一数据选择器74LS153、八选一数据选择器74LS151等。

双四选一数据选择器74LS153逻辑框图如图4.3.45(a)所示。图中，A_0A_1为两个四选一数据选择器的公共地址输入端。$D_{10}\sim D_{13}$、Y_1为上面的四选一数据选择器的输入输出；$D_{20}\sim D_{23}$、Y_2为下面的四选一数据选择器的输入输出；外部输入$\overline{S}_1$、$\overline{S}_2$为两个四选一数据选择器的使能端。

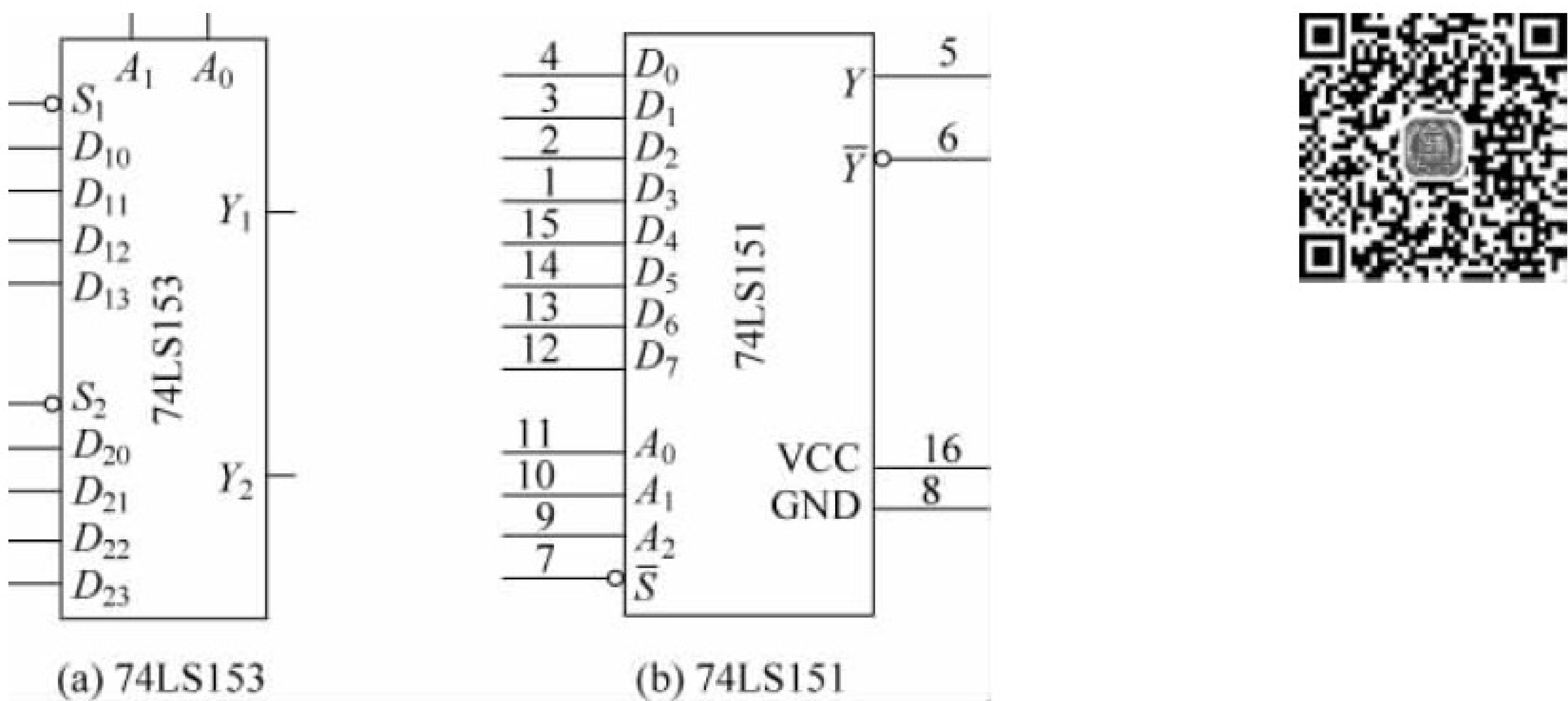


图4.3.45 数据选择器逻辑框图

当$\overline{S}_1=0$时，上面的四选一数据选择器工作，Y_1为有效输出；当$\overline{S}_2=0$时，下面的四选一数据选择器工作，Y_2为有效输出。

QuartusⅡ中库元件74153原理图如图4.3.46所示。图中，A、B对应地址输入端A_1A_0；1C0～1C3、1Y，2C0～2C3、2Y为上、下两个四选一数据选择器的输入输出；1GN、2GN为两个四选一数据选择器的使能端。

由图可看出，当A=B=2GN=0时，与2C0输入端相连的四输入与门的另外3根输入线均为高电平，其他的3个四输入与门的另外3个非外部输入线至少有1根为低电平，2Y=2C0。有兴趣的读者可进一步分析该电路。

由图可看出，由于上、下两个四选一数据选择器共用一组地址输入端，因此，74153并非两个独立的四选一数据选择器，常把它作为一个四选一数据选择器或扩展为一个八选一数据选择器使用。

八选一数据选择器74LS151的引脚功能如图4.3.45(b)所示。74LS151的真值表如表4.3.9所示。它有8个数据输入端$D_0\sim D_7$、3个地址输入端$A_0\sim A_2$、1个选通控制端$\overline{S}$、2个互补的输出端Y和$\overline{Y}$。当选通控制端$\overline{S}=1$时，选择器被禁止，即不工作($Y=0$)。此时输入的数据和地址信号均不起作用。当选通控制端$\overline{S}=0$时，选择器工作，输出Y的逻辑表

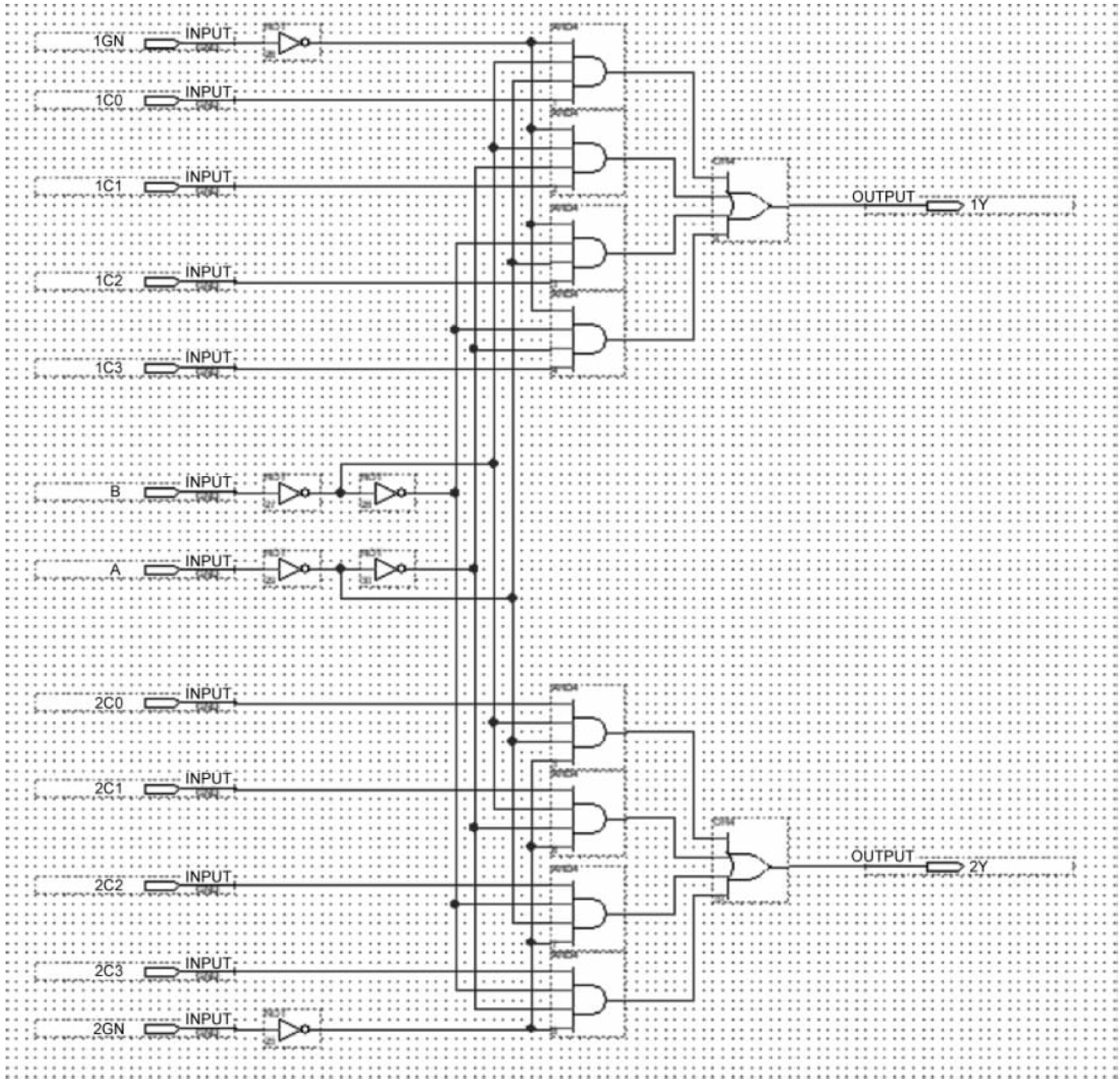


图 4.3.46 Quartus Ⅱ 中库元件 74153 原理图

达式为

$$Y = D_0\overline{A}_2\overline{A}_1\overline{A}_0 + \cdots + D_7 A_2 A_1 A_0 = \sum_{i=0}^{7} D_i m_i \tag{4.3.9}$$

表 4.3.9 74LS151 真值表

$\overline{S}$	$A_2\ \ A_1\ \ A_0$	Y
1	× × ×	0
0	0 0 0	D_0
0	m_i	D_i

当一片数据选择器不能满足应用要求时,可用多片扩展。可通过下面的例题来进一步理解。



【例 4.3.4】 试用 74LS153 构成一个八选一数据选择器。

解 如图 4.3.47(a)所示是用 74LS153 结合门电路构成的八选一数据选择器，简要分析如下。

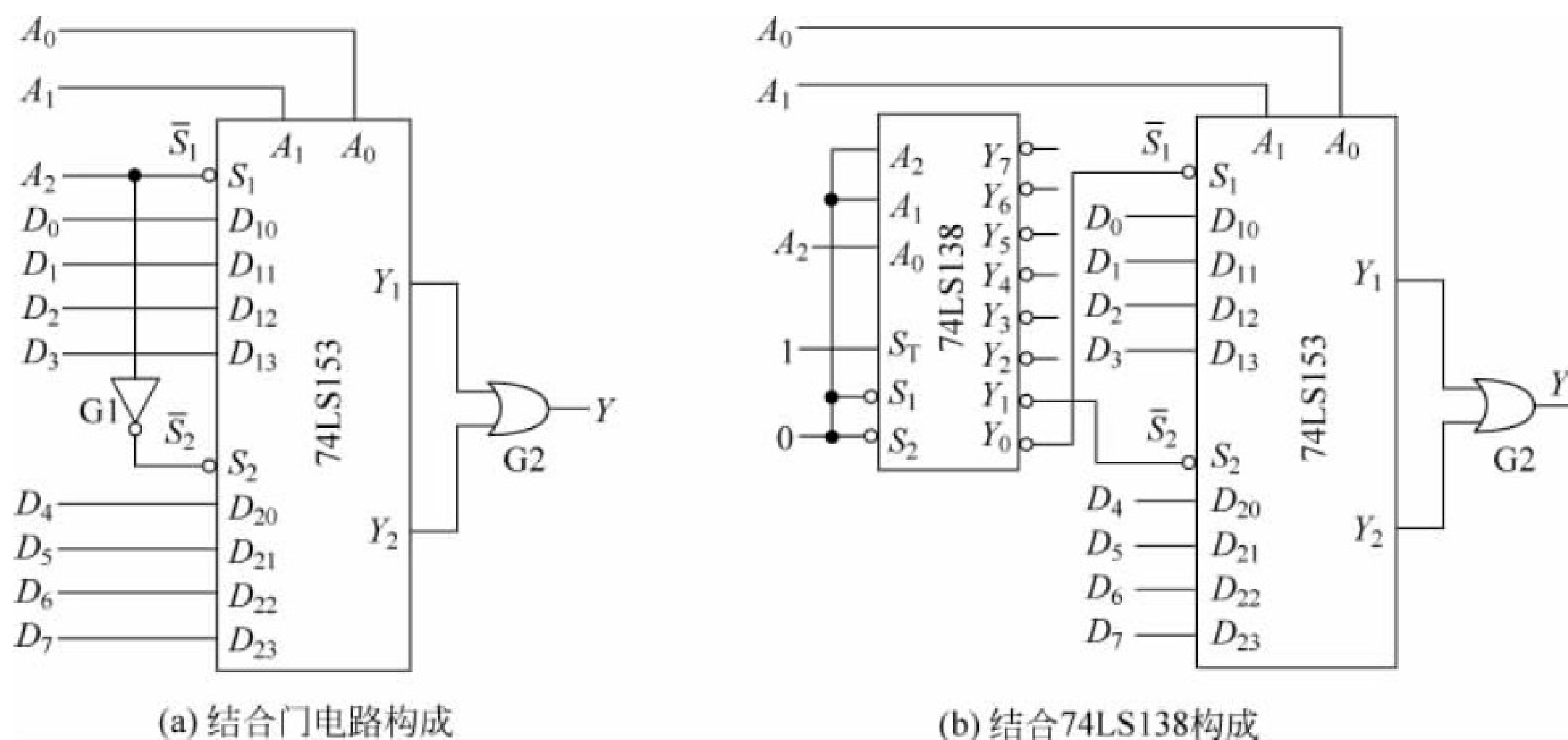


图 4.3.47 例 4.3.4 的图

(1) 输入定义。74LS153 具有 8 路外部输入。令上面的四选一数据选择器的 4 路外部输入对应 D_0～D_3，下面的四选一数据选择器的 4 路外部输入对应 D_4～D_7。

(2) 地址输入控制引脚的连接。令 A_1、A_0 接 74LS153 的 A_1、A_0。根据上面的输入定义关系，有：$A_2=0$，上面的四选一数据选择器工作；$A_2=1$，下面的四选一数据选择器工作。

按照逻辑功能要求，可令 A_2接$\bar{S}_1$，取反后接$\bar{S}_2$。

(3) 输出的连接。依照逻辑功能，当 Y_1输出有效时，$Y=Y_1$；当 Y_2输出有效时，$Y=Y_2$；当 Y_1、Y_2均为原码输出时，有 $Y=Y_1+Y_2$。

如图 4.3.47(b)所示为用 74LS153 结合 74LS138 译码器构成八选一数据选择器。

图 4.3.47(b)中，输入、输出连接方式与图 4.3.47(a)相同，主要区别在于地址输入控制引脚的连接不同。

图中，74LS138 控制端接成正常译码方式；地址端 A_0接 A_2，其余接 0。当 $A_2=0$ 时，74LS138 的 Y_0输出有效，$\bar{S}_1=Y_0=0$，$\bar{S}_2=Y_1=1$，上面的四选一数据选择器工作，下面的四选一数据选择器不工作。当 $A_2=1$ 时，Y_1输出有效，$\bar{S}_1=Y_0=1$，$\bar{S}_2=Y_1=0$，下面的四选一数据选择器工作，上面的四选一数据选择器不工作。

由如图 4.3.47(b)所示电路可以看出，译码器的输出哪个有效取决于所给的地址，非常适合于多个芯片的级联应用。如图 4.3.48 所示电路为四片八选一数据选择器、一个 3-8 译码器和一个或门构成的三十二选一数据选择器。图中，2-4 译码器(用 3-8 译码器实现)对输入的地址 A_4A_3进行译码，其输出 $\bar{Y}_0$～$\bar{Y}_3$ 作为选通控制信号分别接到 4 个八选一数据选择器的 $\bar{S}$ 端。例如，当 $A_4A_3=00$ 时，译码器输出 $\bar{Y}_0=0$，其余各输出端为 1，因此，只有数据选择器片 1 被选通，在 A_2～A_0地址码的作用下，从输入的数据 D_0～D_7中选择一路输出；类

似当 $A_4A_3=01$ 时，$\overline{Y}_1=0$，数据选择器片 2 被选通，在 $A_2 \sim A_0$ 地址码的作用下，从输入的数据 $D_8 \sim D_{15}$ 中选取一路输出；等等。具体讲，若已知地址码 $A_4A_3A_2A_1A_0=01110$，则译码器输出 $\overline{Y}_1=0$，数据选择器片 2 被选通，$Y=Y_2=(D)_{26}=D_{14}$，即选中第 14 路数据 D_{14} 作为输出。

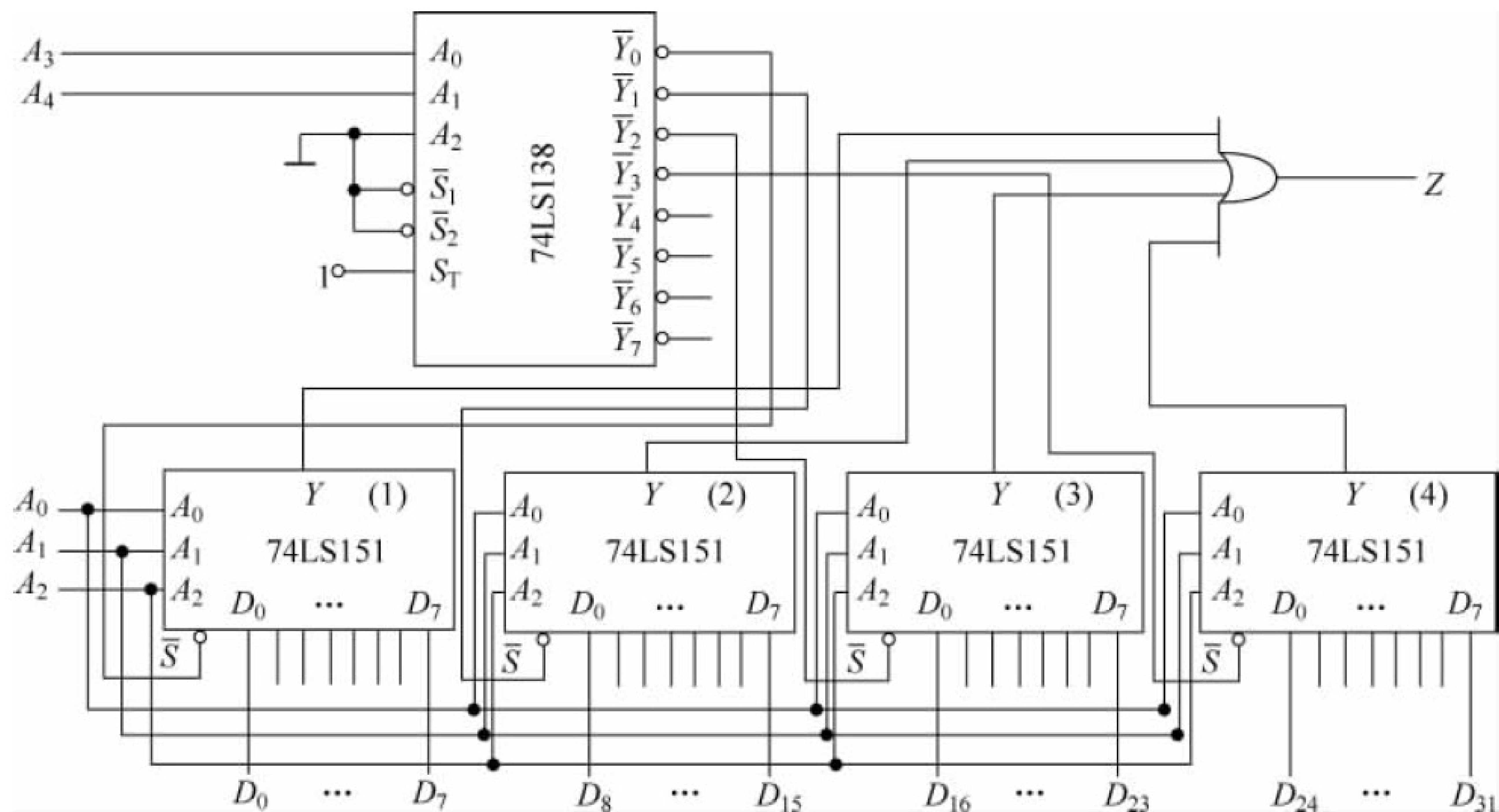


图 4.3.48　三十二选一数据选择器

4.3.5　数值比较器

在数字电路中，经常需要对两个位数相同的二进制数进行比较，以判断它们的相对大小或者是否相等，用来实现这一功能的逻辑电路称为数值比较器。

依照数值比较器的逻辑功能描述，数值比较器输入包括 A、B 两个数，比较结果有 $A>B$、$A<B$、$A=B$ 三种情况。

当 A 和 B 都是 1 位数时，该比较器称为 1 位数值比较器。可写出 1 位数值比较器的真值表，如表 4.3.10 所示。

表 4.3.10　1 位数值比较器

A	B	$Y_{A>B}$	$Y_{A<B}$	$Y_{A=B}$
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

由真值表得到如下逻辑表达式：

$$Y_{A>B}=A\overline{B}$$

$$Y_{A<B}=\overline{A}B$$

$$Y_{A=B}=\overline{A}\overline{B}+AB \tag{4.3.10}$$

读者可根据上面的逻辑表达式直接画出1位数值比较器的逻辑电路。

1位数值比较器是多位比较器的基础。当进行多位数值比较时，先进行高位的比较，当高位比较结果为相等时，再进行低位数值的比较。如2位数值比较器，当 $A_1>B_1$ 或者 $A_1=B_1$ 但 $A_0>B_0$ 时，$A>B$。可参考式(4.3.10)直接写出2位数值比较器 $Y_{A>B}$ 的逻辑表达式如下：

$$Y_{A>B}=A_1\overline{B}_1+(\overline{A}_1\overline{B}_1+A_1B_1)A_0\overline{B}_0 \tag{4.3.11}$$

式中，第1项表示 $A_1>B_1$；第2项表示 $A_1=B_1$，但 $A_0>B_0$。

读者可参考上面的方法进一步总结 $Y_{A=B}$、$Y_{A<B}$ 的逻辑表达式。

如图4.3.49所示为集成的4位数值比较器74LS85的逻辑符号。图中，$A_3A_2A_1A_0$、$B_3B_2B_1B_0$ 为两个4位的二进制数；$Y_{A>B}$、$Y_{A=B}$、$Y_{A<B}$ 为输出的3个比较结果。考虑扩展，74LS85增加了3个来自低位的比较结果输入 $I_{A>B}$、$I_{A=B}$、$I_{A<B}$。

根据4位二进制数相互比较的逻辑关系，当 $A>B$ 时，$Y_{A>B}=1$；当 $A<B$ 时，$Y_{A<B}=1$；当 $A=B$ 时，输出结果由来自低位的比较结果输入 $I_{A>B}$、$I_{A=B}$、$I_{A<B}$ 确定。

用两片74LS85构成的8位数值比较器如图4.3.50所示。图中，令片2的两个4位二进制数分别对应两个8位二进制数据中的高4位。片1的两个4位二进制数分别对应两个8位二进制数据中的低4位。低4位没有来自低位的比较结果，令该芯片的 $I_{A=B}$ 有效，$I_{A>B}$、$I_{A<B}$ 无效。

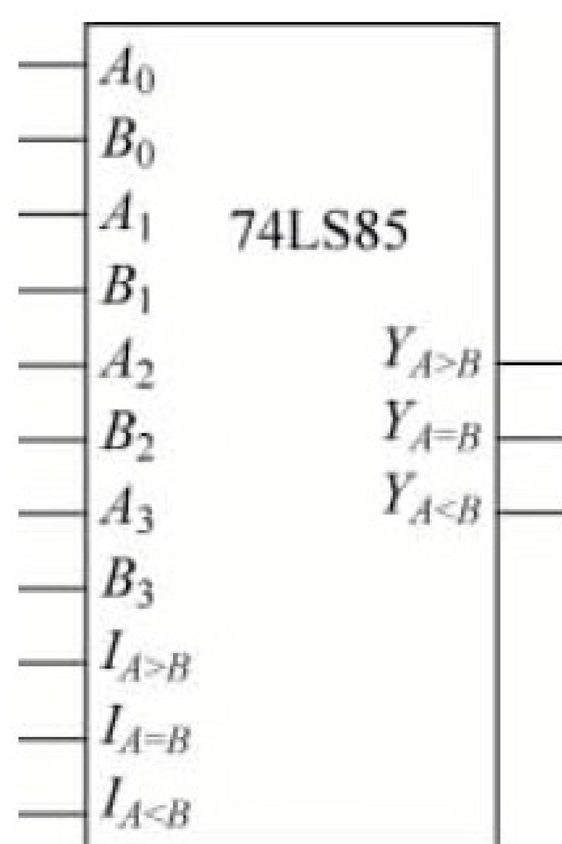


图4.3.49 74LS85

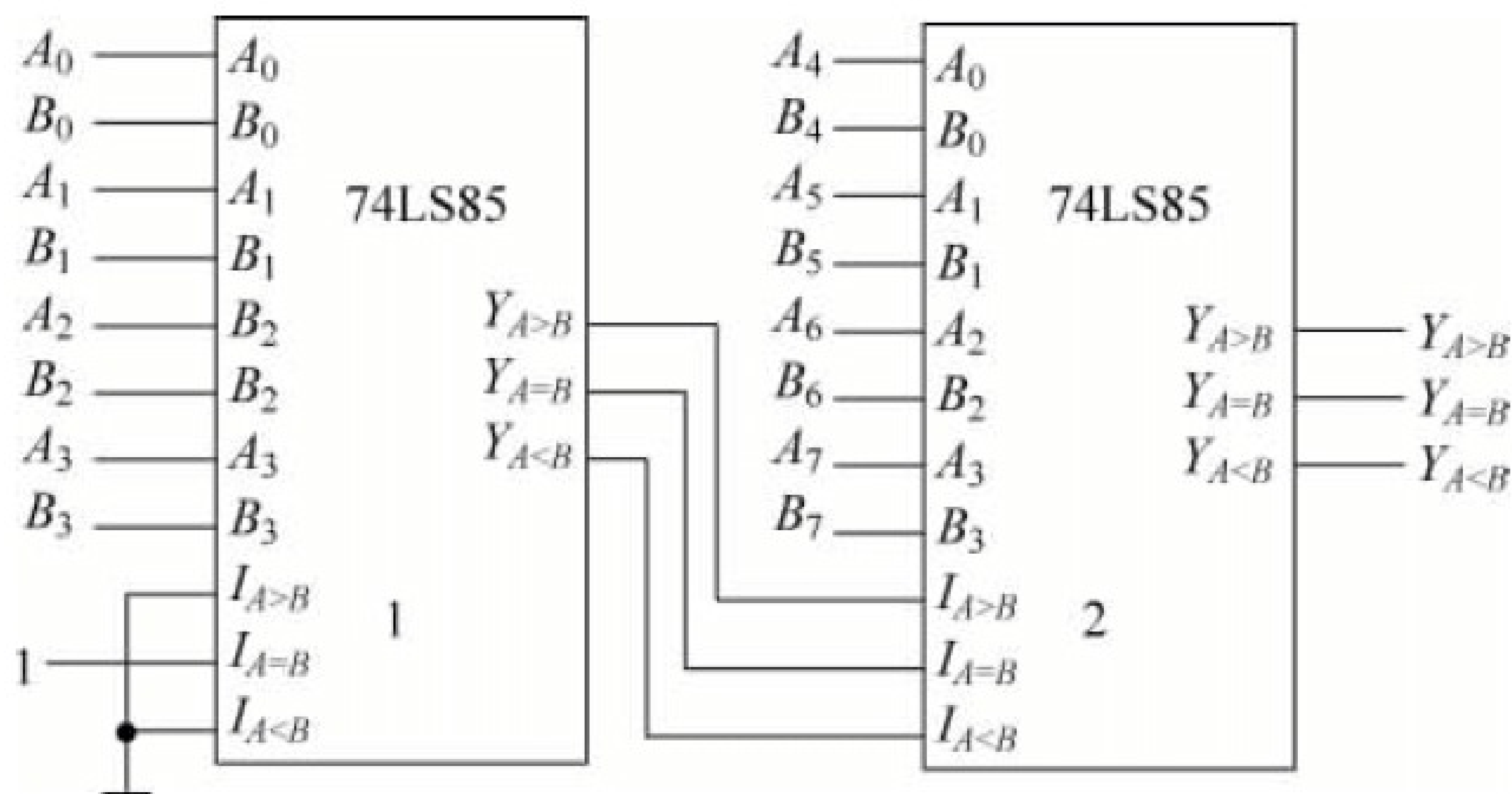


图4.3.50 用74LS85构成8位数值比较器

由74LS85的逻辑功能，当片2的两个4位二进制数可直接比较出大于(或小于)关系时，直接给出输出 $Y_{A>B}$(或 $Y_{A<B}$)。当片2的两个4位二进制数相等时，根据片1的低4位数据比较结果确定最终的比较结果，实现了8位数值比较器的逻辑功能。

复习与思考

4.3.1 根据你的理解谈谈如图4.3.4所示的3位二进制普通编码器有无实用意义。

4.3.2 已知某逻辑问题具有 A、B 两个输入。当 A 有效时，输出 $Y=A$；当 B 有效时，输出 $Y=B$，求解函数 Y 的表达式。

4.3.3 尝试用不同于教材的解法求解例4.3.2。

4.3.4 总结二进制译码器与二—十进制译码器逻辑功能的异同。

4.4 利用中规模器件实现组合逻辑电路(MSI 设计)

译码器、数据选择器的输出包括了其地址输入变量对应的全部最小项,可利用译码器、数据选择器的输出实现各种组合逻辑电路。



4.4.1 用译码器实现组合逻辑电路

当二进制的3-8译码器控制端 $S_T=1$、$\bar{S}_1=\bar{S}_2=0$ 时,如果将地址端 A_2、A_1、A_0 作为3个输入的自变量,则8个输出端输出的就是这3个输入变量的全部最小项,即 $\bar{Y}_i=\bar{m}_i$。利用附加的门电路将这些最小项适当地组合起来,便可产生任何形式的三变量组合逻辑函数。以此类推,n 位二进制译码器的输出给出了 n 变量的全部最小项,利用附加的门电路可获得任何形式输入变量数不大于 n 的组合逻辑函数。

【例 4.4.1】 试用3-8译码器74LS138设计能实现下列多输出函数的组合逻辑电路。输出的逻辑函数为

$$\begin{cases} Z_1 = \bar{A}\bar{B}\bar{C} + ABC \\ Z_2 = AB + BC + AC \\ Z_3 = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC \end{cases} \tag{4.4.1}$$

解 (1) 先将输出函数变换为标准最小项表达式,有

$$\begin{cases} Z_1 = \bar{A}\bar{B}\bar{C} + ABC = m_0 + m_7 \\ Z_2 = AB + BC + AC = AB(C+\bar{C}) + (A+\bar{A})BC + A(B+\bar{B})C \\ \quad = ABC + AB\bar{C} + ABC + \bar{A}BC + ABC + A\bar{B}C \\ \quad = ABC + AB\bar{C} + \bar{A}BC + A\bar{B}C = m_3 + m_5 + m_6 + m_7 \\ Z_3 = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC = m_1 + m_2 + m_4 + m_7 \end{cases} \tag{4.4.2}$$

(2) 设置逻辑函数的输入与译码器的输入一致。令74LS138译码器的地址端分别为 $A_2=A$、$A_1=B$、$A_0=C$,则它的输出就是式(4.4.2)中的 $\bar{m}_0 \sim \bar{m}_7$。

(3) 进一步变换逻辑函数为译码器输出 $\bar{Y}_i(\bar{m}_i)$ 的组合,即

$$Z_1 = m_0 + m_7 = \overline{\bar{m}_0 \cdot \bar{m}_7} = \overline{\bar{Y}_0 \cdot \bar{Y}_7}$$

$$Z_2 = m_3 + m_5 + m_6 + m_7 = \overline{\bar{m}_3 \cdot \bar{m}_5 \cdot \bar{m}_6 \cdot \bar{m}_7} = \overline{\bar{Y}_3 \cdot \bar{Y}_5 \cdot \bar{Y}_6 \cdot \bar{Y}_7}$$

$$Z_3 = m_1 + m_2 + m_4 + m_7 = \overline{\bar{m}_1 \cdot \bar{m}_2 \cdot \bar{m}_4 \cdot \bar{m}_7} = \overline{\bar{Y}_1 \cdot \bar{Y}_2 \cdot \bar{Y}_4 \cdot \bar{Y}_7}$$

(4) 画出电路图。由第(3)步的公式画出实现函数 Z_1、Z_2、Z_3 的组合逻辑电路如图4.4.1所示,画电路图时应将译码器接成正常译码的工作模式。

用译码器实现组合逻辑电路的步骤如下:

(1) 根据给定的条件求出待求逻辑函数的最小项表达式。

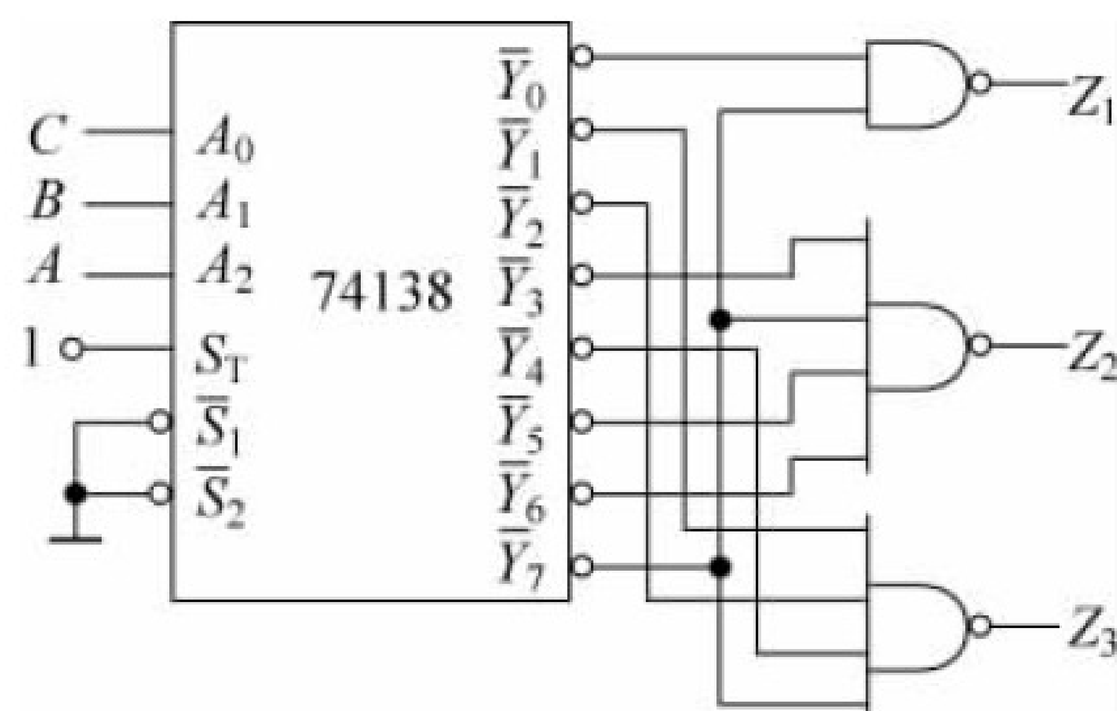


图 4.4.1 例 4.4.1 的电路图

(2) 设置逻辑函数的输入与译码器的输入一致。

(3) 变换逻辑函数为译码器输出 $\bar{Y}_i(\bar{m}_i)$的组合。

(4) 根据逻辑函数的译码器的输出组合表达式画出电路图,画电路图时应将译码器接成正常译码的工作模式。

【例 4.4.2】 试用 3-8 译码器实现两个 1 位二进制数的全减运算。

解 (1) 分析逻辑功能要求,写出全减运算的真值表并求出标准与或式。

设 A_i、B_i、C_{i-1}分别表示被减数、减数、低一位的借位数。F_i、C_i表示差值,向高一位的借位数。根据全减运算的功能要求,可以写出如表 4.4.1 所示的两个 1 位二进制的全减真值表。

表 4.4.1 例 4.4.2 的真值表

A_i	B_i	C_{i-1}	F_i	C_i
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

由真值表写出逻辑函数 F_i、C_i的标准与或式:

$$\begin{cases} F_i = m_1 + m_2 + m_4 + m_7 \\ C_i = m_1 + m_2 + m_3 + m_7 \end{cases} \tag{4.4.3}$$

(2) 设置逻辑函数的输入与译码器的输入一致。令 74LS138 译码器的地址端分别为 $A_2=A_i$、$A_1=B_i$、$A_0=C_{i-1}$,则它的输出就是式(4.4.3)中的$\bar{m}_0\sim\bar{m}_7$。

(3) 进一步变换逻辑函数为译码器输出 $\bar{Y}_i(\bar{m}_i)$的组合,即

$$\begin{cases} F_i = m_1 + m_2 + m_4 + m_7 = \overline{\bar{m}_1\,\bar{m}_2\,\bar{m}_4\,\bar{m}_7} \\ C_i = m_1 + m_2 + m_3 + m_7 = \overline{\bar{m}_1\,\bar{m}_2\,\bar{m}_3\,\bar{m}_7} \end{cases} \tag{4.4.4}$$

(4) 画电路。由式(4.4.4)可以知道,增加两个与非门就可以实现函数 F_i、C_i。如图 4.4.2 所示为实现两个 1 位二进制数的全减运算的电路。

类似地,也可以选用 4-10 BCD8421 译码器,实现两个 1 位二进制数的全减运算。

有兴趣的读者可参考图 4.4.2 进一步设计该电路。

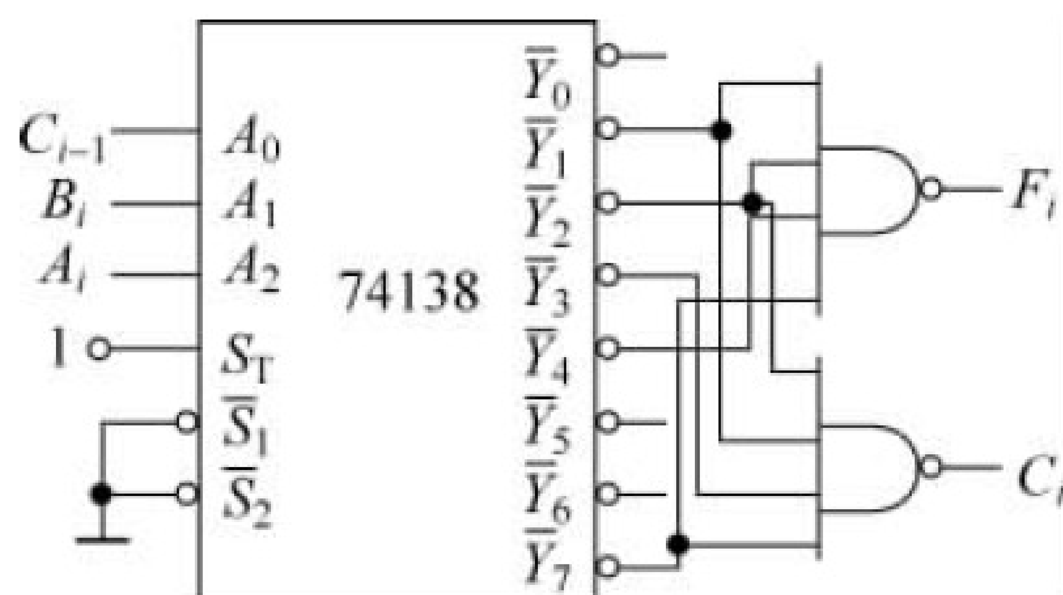


图 4.4.2 例 4.4.2 的电路图

4.4.2 用数据选择器实现组合逻辑电路

当 $\overline{S}=0$ 时,八选一数据选择器的输出表达式,可以写成

$$Y = D_0\overline{A}_2\overline{A}_1\overline{A}_0 + \cdots + D_7A_2A_1A_0 = \sum_{i=0}^{7} D_i m_i \qquad (4.4.5)$$

如果用地址端 A_2、A_1、A_0 分别代表 3 个变量 A、B、C,适当地选择 $D_0 \sim D_7$,可以用八选一数据选择器设计任意的 3 变量组合逻辑电路。

从实现技巧角度,可采用卡诺图对照、表达式对照等多种方式来设计实现组合逻辑电路。

1. 用卡诺图对照实现组合逻辑电路

式(4.4.5)用卡诺图的形式表示如图 4.4.3 所示。可求出待求逻辑问题的卡诺图,对照图 4.4.3 选择 $D_0 \sim D_7$ 的值,使八选一数据选择器的卡诺图与待求逻辑问题的卡诺图相同,则二者具有相同的逻辑功能。

A_2 \ A_1A_0	00	01	11	10
0	D_0	D_1	D_3	D_2
1	D_4	D_5	D_7	D_6

图 4.4.3 数据选择器卡诺图

用卡诺图对照实现组合逻辑电路步骤如下:

(1) 根据给定的条件求出待求逻辑问题的卡诺图。

(2) 设置待求逻辑问题的输入与数据选择器的输入一致。

(3) 将待求逻辑问题的卡诺图与数据选择器的卡诺图进行对照,求出 $D_0 \sim D_7$ 的值。

(4) 根据 $D_0 \sim D_7$ 的值画出电路图(应将数据选择器的使能端 $\overline{S}$ 接地)。

【例 4.4.3】 试利用八选一数据选择器,设计一个三变量的判偶电路。

解 (1) 求出待求逻辑问题的卡诺图。

① 做出真值表。根据 3 个变量 A、B、C 判偶的逻辑功能,可写出如表 4.4.2 所示的真值表(1 为有效输入)。

表 4.4.2　例 4.4.3 的真值表

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

② 做卡诺图。由判偶函数 F 的真值表作出如图 4.4.4 所示的卡诺图。

(2) 设置待求逻辑问题的输入与数据选择器的输入一致。令 $A=A_2$、$B=A_1$、$C=A_0$，使待求逻辑问题的输入与数据选择器的输入一致。

(3) 对照两个卡诺图，求出 $D_0 \sim D_7$的值。将图 4.3.4 与图 4.3.3 所示数据选择器的卡诺图进行比较，则有 $D_0=D_3=D_5=D_6=1$，$D_1=D_2=D_4=D_7=0$。

(4) 画出电路图。可画出如图 4.4.5 所示的三变量判偶电路图。

A \ BC	00	01	11	10
0	1		1	
1		1		1

图 4.4.4　例 4.4.3 的卡诺图

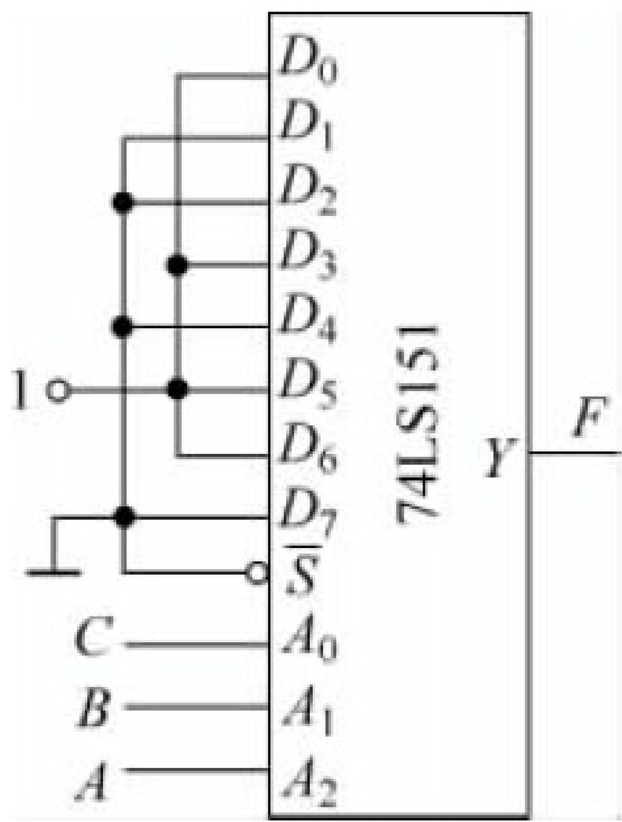


图 4.4.5　最终电路图

从上面的例子可以看出，用具有 n 个地端输入端的数据选择器设计 $m=n$ 变量函数的组合逻辑电路，是十分方便的。它不需要将所设计的函数化简为最简式，只需将输入变量加到数据选择器的地址端，选择器的数据输入端按卡诺图中最小项格中的值(0 或 1)对应相连即可。

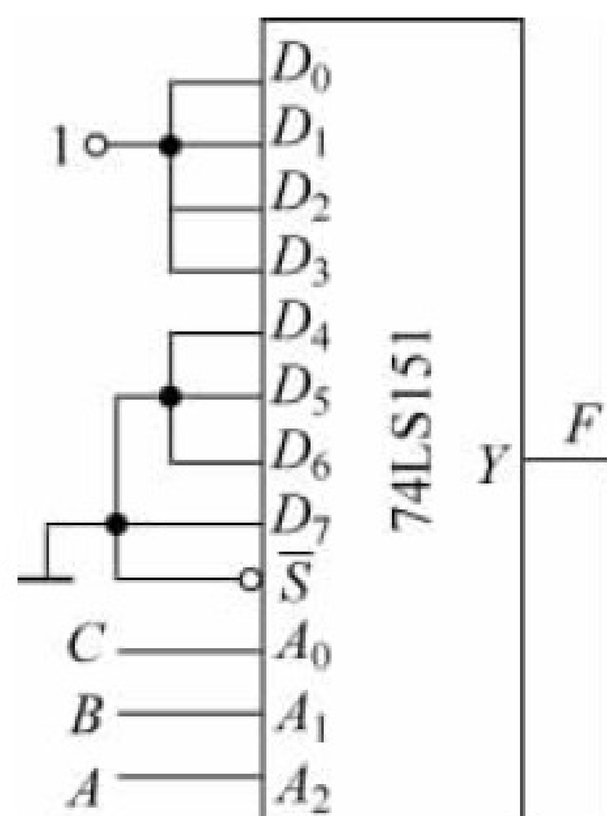
类似地，当分析用数据选择器构成的组合逻辑电路时，可直接根据输入地址连接关系确定输入逻辑变量，根据数据输入端的连接关系直接求出卡诺图，进而求出该电路对应的逻辑函数。根据这个思路，如图 4.4.6 所示的电路中，A_2、A_1、A_0分别接外部输入为 A、B、C，数据输入端 $D_0=D_1=D_2=D_3=1$，可画出卡诺图如图 4.4.7 所示。化简卡诺图，可求出该电路的逻辑函数

$$F=\overline{A}$$

当输入变量数小于数据选择器的地址端数(即 $n>m$)时，例如用八选一数据选择器设

计二变量函数的组合逻辑电路时，只需将高位地址端 A_2 接地以及相应的数据输入端（D_4～D_7）接地即可实现。如图 4.4.8 实例 2 所示电路中的数据输入端只有 D_3 为 1。有

$$Y = D_3\overline{A}_2A_1A_0 = D_3\overline{0}AB = AB$$



A \ BC	00	01	11	10
0	1	1	1	1
1				

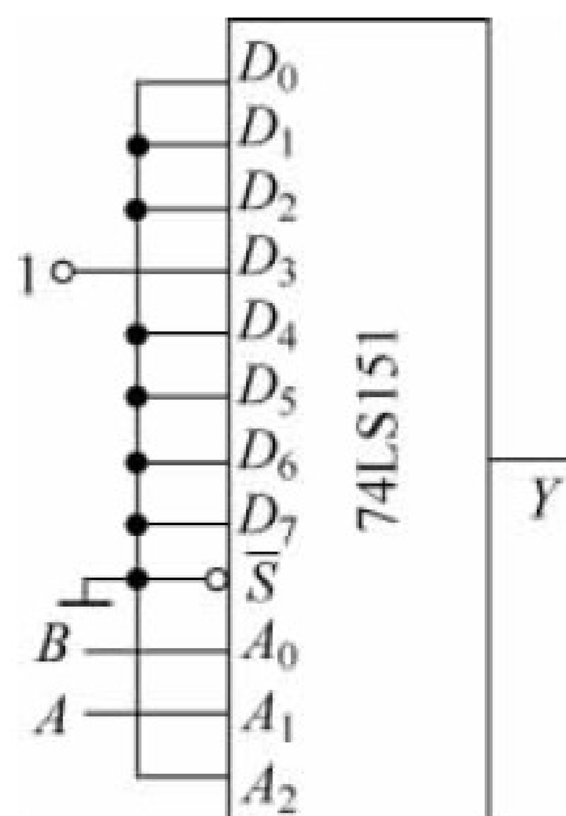


图 4.4.6　实例 1　　图 4.4.7　实例 1 的卡诺图　　图 4.4.8　实例 2

2. 用具有 n 个地址端的数据选择器实现 m（$m>n$）变量的逻辑函数



当 $m>n$ 时，可以将 2^n 选 1 数据选择器扩展成 2^m 选 1 数据选择器，然后按照上面的方法予以设计。

如图 4.4.9 所示为用具有 3 个地址端的八选一数据选择器扩展成十六选一数据选择器来实现四变量逻辑函数的实例，这种方法称为扩展法。

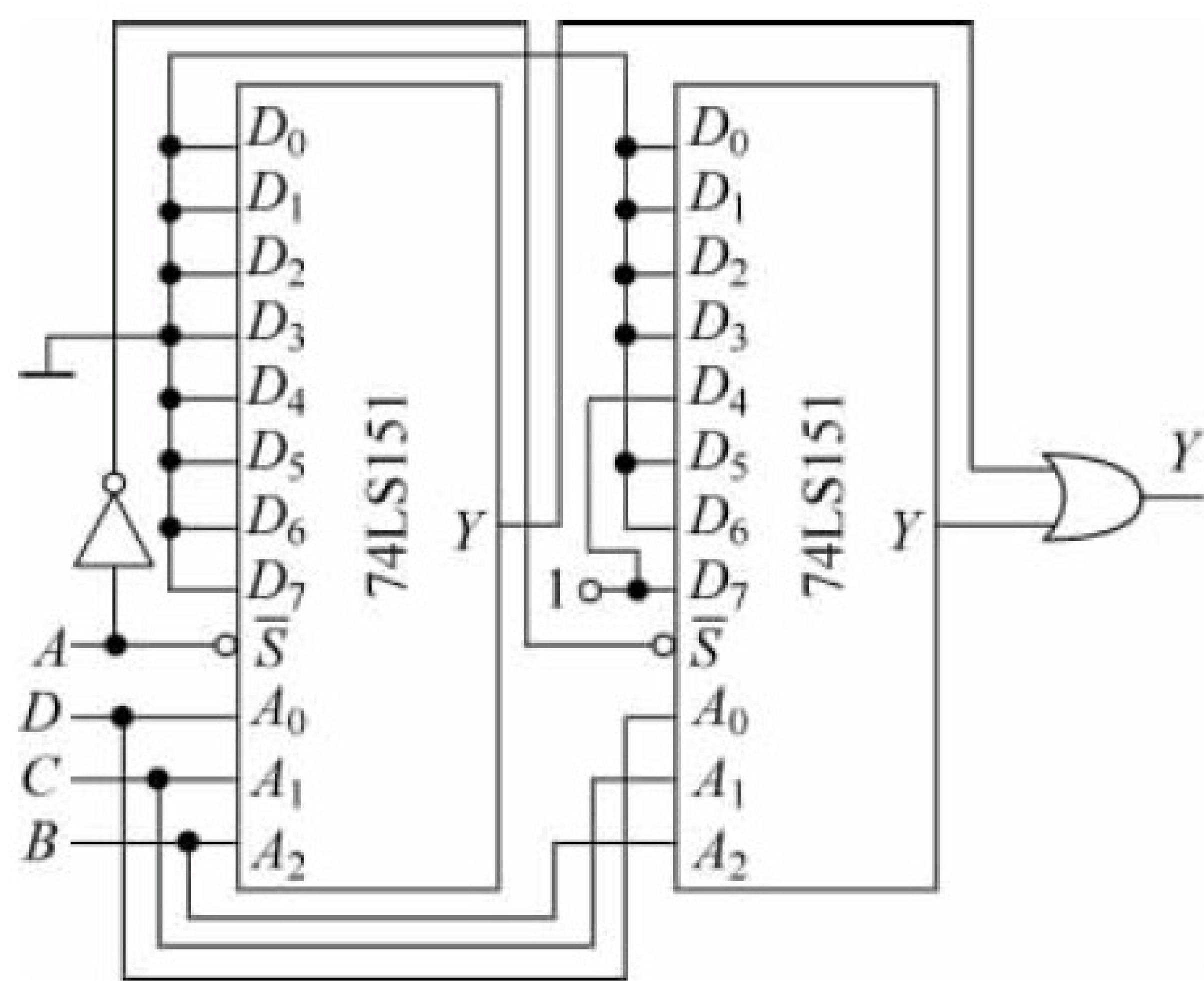


图 4.4.9　扩展法实例

图 4.4.9 中，外部输入 A 接左边选择器的使能端 $\overline{S}$，取反后接右边选择器的使能端 $\overline{S}$。当 $A=0$ 时，左边的选择器工作，右边的选择器不工作。当 $A=1$ 时，右边的选择器工作，左边的选择器不工作。两个选择器整体构成十六选一数据选择器。

图 4.4.9 中，右边的选择器中的 $D_4=D_7=1$，相当于十六选一选择器的 $D_{12}=D_{15}=1$，可写出如图 4.4.9 所示电路的逻辑函数为

$$Y = ABCD + AB\overline{C}\overline{D}$$

当 $m=n+1$ 时，也可用 2^n 选 1 数据选择器直接完成设计。下面采用表达式对照结合实

例介绍实现方法。用表达式对照实现组合逻辑电路步骤如下：

(1) 根据给定的条件求出待求逻辑问题的标准表达式。

(2) 设置待求逻辑问题的输入与数据选择器的输入一致。

(3) 将待求逻辑问题的表达式与数据选择器的输出表达式进行对照，求出 $D_0 \sim D_{n-1}$ 的值。

(4) 根据 $D_0 \sim D_{n-1}$ 的值画出电路图。

当 $m=n+1$ 时，应对待求逻辑问题的表达式进行适当合并处理，使其 n 个变量对应的最小项无重复。

【例 4.4.4】 试利用四选一数选器实现函数 $Y=AB+AC+BC$。

解 (1) 根据给定的条件求出待求逻辑问题的标准表达式。

$$Y = AB + AC + BC = AB(C+\bar{C}) + A(B+\bar{B})C + (A+\bar{A})BC$$
$$= ABC + AB\bar{C} + A\bar{B}C + \bar{A}BC$$

(2) 设置待求逻辑问题的输入与数据选择器的输入一致。

令逻辑函数 $A=A_1$、$B=A_0$，有

$$Y = ABC + AB\bar{C} + A\bar{B}C + \bar{A}BC = A_1A_0C + A_1A_0\bar{C} + A_1\bar{A}_0C + \bar{A}_1A_0C$$

合并有重复的最小项，有

$$Y = A_1A_0 + A_1\bar{A}_0C + \bar{A}_1A_0C \tag{4.4.6}$$

(3) 对照两个表达式，求出 $D_0 \sim D_3$ 的值。四选一数据选择器输出表达式为

$$Y = D_3A_1A_0 + D_2A_1\bar{A}_0 + D_1\bar{A}_1A_0 + D_0\bar{A}_1\bar{A}_0 \tag{4.4.7}$$

对照式(4.4.6)、式(4.4.7)，有

$$D_3 = 1, \quad D_1 = D_2 = C, \quad D_0 = 0$$

(4) 根据 $D_0 \sim D_3$ 的值画出电路图。可画出函数 $Y=AB+AC+BC$ 的逻辑图如图 4.4.10 所示。

显然，本题有多种解法，可令逻辑函数 $A=A_1$，$C=A_0$，有

$$Y = A_1BA_0 + A_1B\bar{A}_0 + A_1\bar{B}A_0 + \bar{A}_1BA_0 = BA_1A_0 + BA_1\bar{A}_0 + B\bar{A}_1A_0$$

对照四选一数据选择器输出表达式，有

$$D_3 = D_1 = D_2 = B, \quad D_0 = 0$$

可画出逻辑图如图 4.4.11 所示。

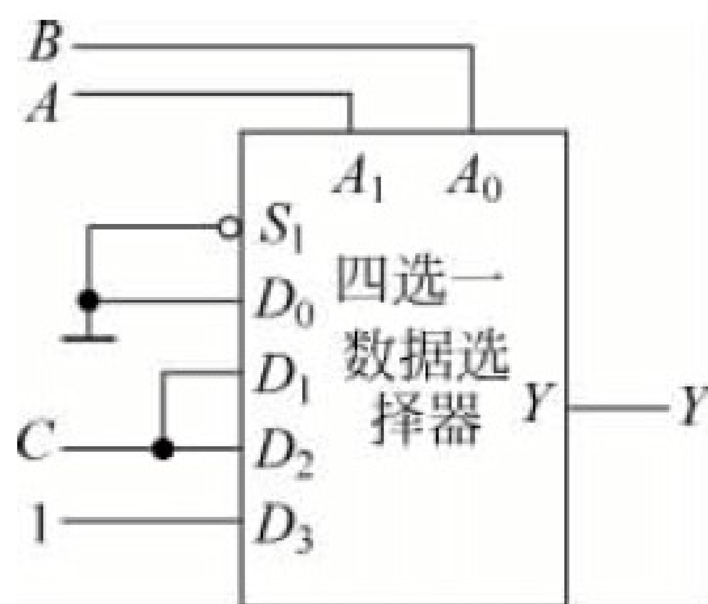


图 4.4.10 例 4.4.4 的图 1

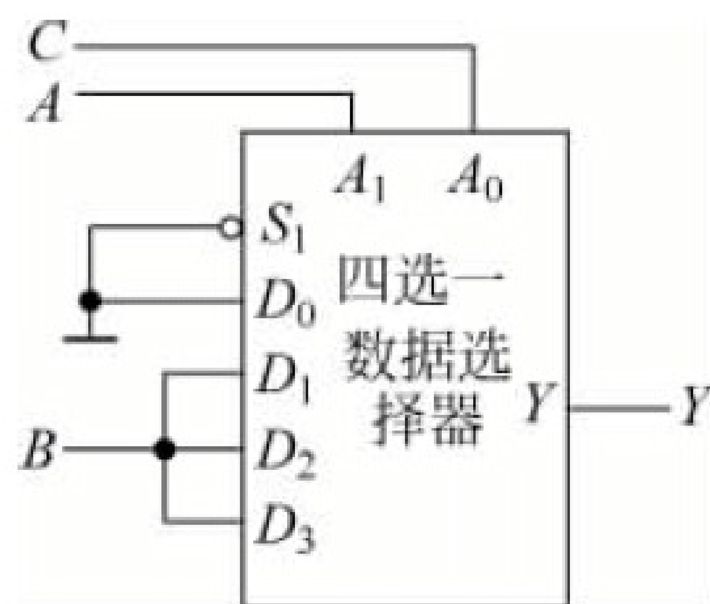


图 4.4.11 例 4.4.4 的图 2

从上述例子中可以看出，使用译码器和附加逻辑门设计多输出函数的组合逻辑电路极为方便；而应用数据选择器设计单输出函数的组合逻辑电路极为方便。

此外，常用的中规模组合逻辑电路通用性强，可在不用或尽量少用附加电路的情况下，将若干功能部件扩展为位数更多、功能更复杂的电路。常用的中规模组合逻辑电路芯片内部一般都设置有缓冲门和使能端（即控制端），使能端除了本身的用途外，还可以用来消除冒险现象。因此，与利用门电路等小规模器件实现组合逻辑电路相比，利用中规模器件实现组合逻辑电路具有设计简单、可靠性强等诸多优点，成为目前组合逻辑电路设计的主流方式之一。

复习与思考

4.4.1　根据你的理解谈谈利用门电路实现组合逻辑和利用译码器等中规模器件实现组合逻辑电路等不同组合逻辑电路设计方法的利弊。

4.4.2　尝试用不同于教材的解法求解例 4.4.4。

4.5　组合逻辑电路中的竞争-冒险现象

在前面介绍的组合逻辑电路的分析与设计方法中，所有的输入、输出均是稳定的逻辑电平。在工程实践中，输入逻辑电平发生变化的瞬间，可能对输出产生影响，进而影响系统的效率及其性能。



4.5.1　竞争-冒险的含义

把门电路的两个输入端的信号同时向相反的逻辑电平跳变的现象，称为竞争。把因输入端的竞争而导致输出可能产生不应有的尖峰干扰脉冲（又称过渡干扰脉冲）的现象称为竞争-冒险。

将如图 4.5.1(a)所示 A、B 信号加在与门输入端。A 由 1 跳变到 0，B 由 0 跳变到 1，存在竞争。由于 B 首先跳变到 1，这样，在极短的 Δt 的时间内，将出现 A、B 同时为 1 的状态，输出 Y 将产生尖峰正脉冲，或称为电压毛刺。

当然，前面分析组合电路时，习惯上忽略门电路的电压传输时间，上面的输出波形是考虑门电路传输时间所得到的可能波形。显然，图 4.5.1 中，产生冒险的原因是输入信号 A、B 没有同时变化到位。由于输入信号不可能完全同步到达，因此图 4.5.1 所示冒险现象是客观存在的。

类似地，将如图 4.5.1(b)所示 A、B 信号加在或门输入端。A 由 1 跳变到 0，B 由 0 跳变到 1，存在竞争。由于 A 首先跳变到 0，这样，在极短的 Δt 的时间内，将出现 A、B 同时为 0 的状态，输出 Y 将产生尖峰负脉冲，存在着冒险现象。

必须指出，有竞争并不一定会产生冒险现象。将如图 4.5.1(c)所示 A、B 信号加在或

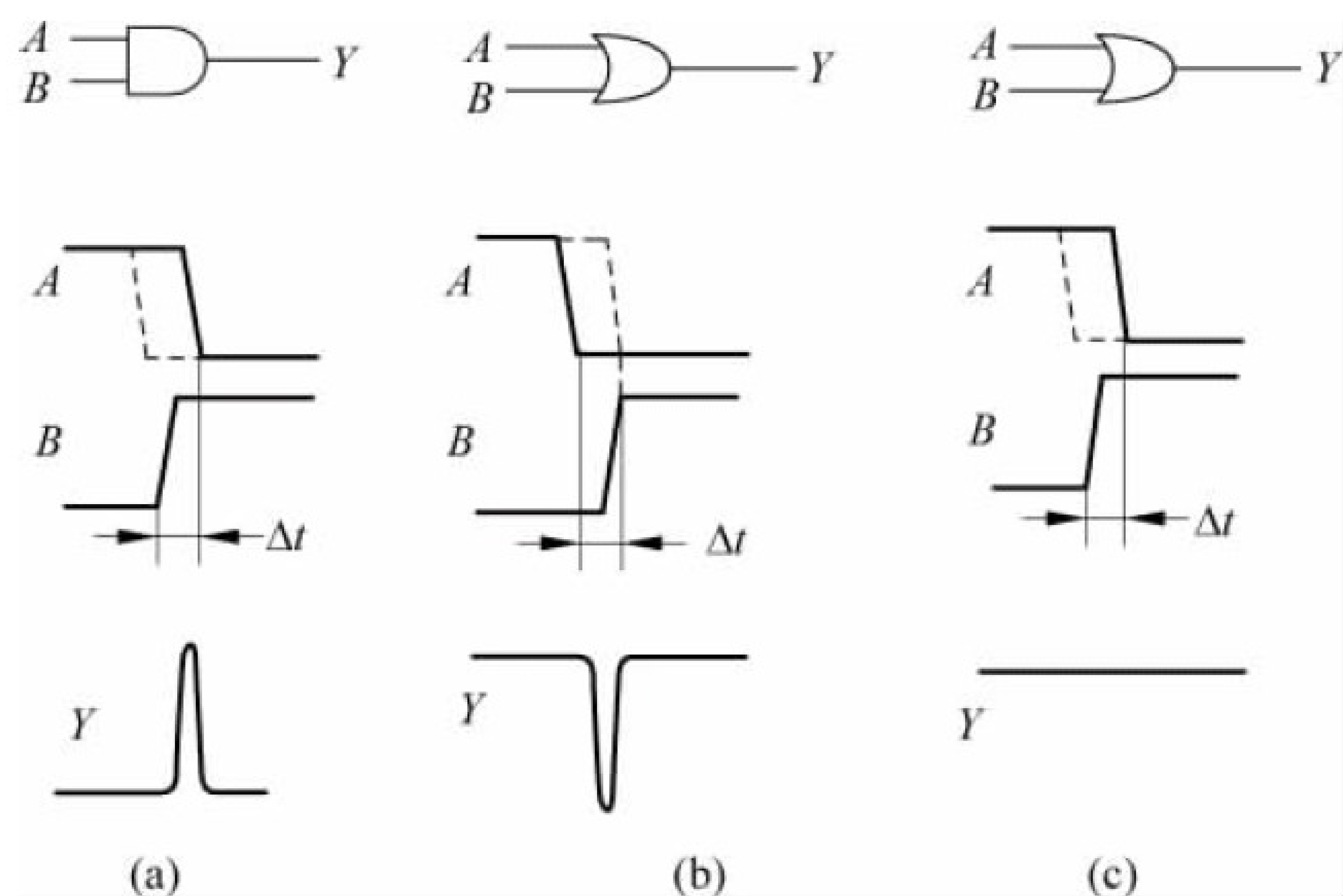


图 4.5.1 竞争-冒险现象

门输入端。A 由 1 跳变到 0，B 由 0 跳变到 1，存在竞争。由于 B 首先跳变到 1，Y 将稳定输出高电平。

4.5.2 检查竞争-冒险现象的方法

图 4.5.1(a)、(b)给出了竞争-冒险的典型例子，图 4.5.1(c)给出了虽然有竞争却不会产生冒险的输入实例。读者可能会问，如图 4.5.1(c)所示电路是否真的存在竞争-冒险？如何检查电路是否存在竞争-冒险的可能？

显然，图 4.5.1(b)证实了如图 4.5.1(c)所示电路存在竞争-冒险。如图 4.5.1(a)、(b)所示电路给出了竞争-冒险的两种典型电路，基于这两个电路，可总结存在竞争-冒险现象的两种典型逻辑函数如下：

$$\begin{cases} Y_1 = A + \overline{A} \\ Y_2 = A \cdot \overline{A} \end{cases} \tag{4.5.1}$$

因为如式(4.5.1)所示函数存在竞争-冒险，因此将输出 Y_1、Y_2取反，可类推出函数

$$\begin{cases} \overline{Y}_1 = \overline{A + \overline{A}} \\ \overline{Y}_2 = \overline{A \cdot \overline{A}} \end{cases} \tag{4.5.2}$$

也存在竞争冒险。当然，由摩根定律，式(4.5.2)和式(4.5.1)具有一致性。

基于上面的分析，可总结通过逻辑函数检查存在竞争-冒险可能的方法：在一定条件下，如果输出端函数可简化为如式(4.5.1)和式(4.5.2)所示函数，可判定该逻辑函数存在竞争-冒险可能。

【例 4.5.1】 试判断如图 4.5.2 所示电路是否存在竞争-冒险现象，已知任何瞬间，输入变量只可能有 1 个改变状态。

解 (1) 图 4.5.2(a)所示电路的逻辑函数为

$$Y = \overline{AB + \overline{B}}$$

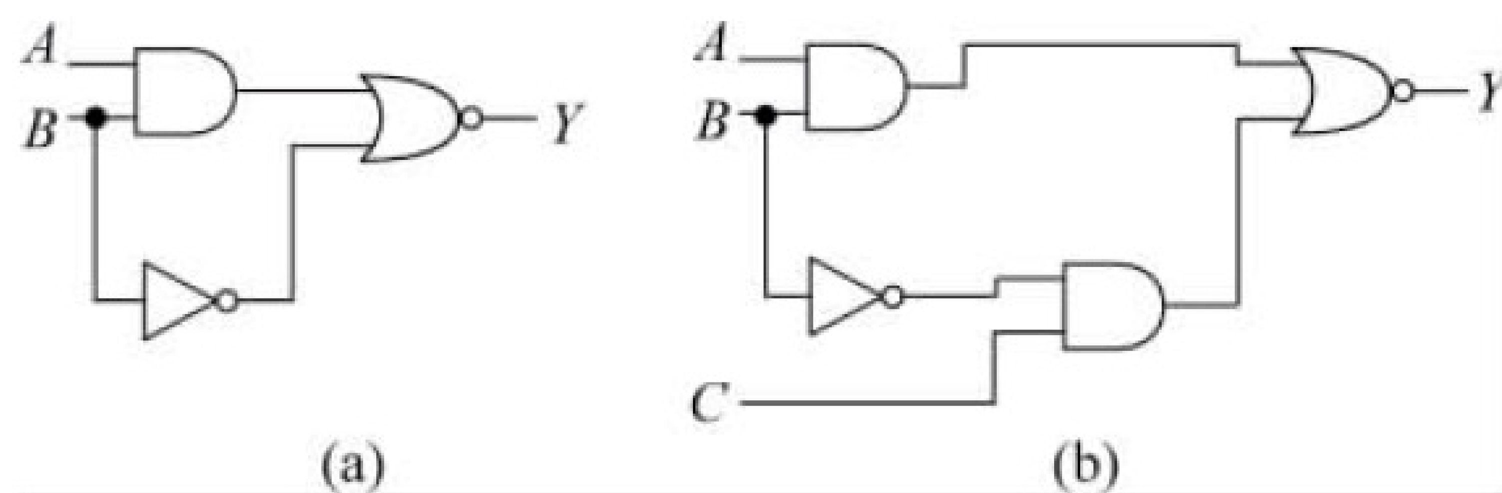


图 4.5.2 例 4.5.1 的图

输入变量只可能有 1 个改变状态，令 $A=1$，B 可改变，有

$$Y=\overline{AB+\overline{B}}=\overline{B+\overline{B}}$$

与式(4.5.2)吻合，存在竞争-冒险现象。该冒险现象与如图 4.5.1(b)所示的现象类似。

(2) 如图 4.5.2(b)所示电路的逻辑函数为

$$Y=\overline{AB+\overline{B}C}$$

输入变量只可能有 1 个改变状态，令 $A=C=1$，B 可改变，有

$$Y=\overline{AB+\overline{B}C}=\overline{B+\overline{B}}$$

与式(4.5.2)吻合，存在竞争-冒险现象。

必须指出，上面的方法虽然简单，但局限性很大。在工程实践中，绝大多数情况下输入变量均是多个变量同时改变的。如果考虑多个变量同时改变，则很难从逻辑表达式上简单找出所有产生竞争-冒险现象，不能找出所有可能性，也就难以得出不存在竞争-冒险现象的结论。

可用实验的方法检查电路的输出端是否竞争-冒险现象。这时，可在电路的输入端尝试所有可能的输入，在输出端观察是否存在尖峰脉冲。此外，还常用计算机辅助分析的方法分析数字电路。必须指出，计算机辅助分析所得到的模拟结果有时和实际电路的工作状态会有一些差异，实验检测的结果才是最终的结论。

4.5.3 消除竞争-冒险现象的方法

1. 接入滤波电容滤出尖峰脉冲

由于尖峰干扰脉冲的宽度一般都很窄(一般在几十纳秒以内)，因此可在可能产生尖峰干扰脉冲的门电路输出端与地之间接入一个很小容量的滤波电容(几十皮法即可)。尖峰干扰脉冲的大部分能量通过电容流向地，可确保尖峰干扰脉冲的幅度削弱至阈值电压以下。

这种方法的优点是简单；弊端是增加了输出电压的上升时间及下降时间，很大程度上使电路的动态特性变差。

2. 引入选通脉冲

对输出可能产生尖峰干扰脉冲的门电路增加一个接选通信号的输入端，只有在输入信

号转换完成并稳定后，才引入选通脉冲将它打开，此时才允许有输出。在转换过程中，由于没有加选通脉冲，因此，输出不会出现尖峰干扰脉冲。

利用选通脉冲，在选通脉冲到来前，提前准备输入信号，可保证稳定状态下无尖峰脉冲。对于具有使能端的集成电路芯片，可提前准备输入信号，利用选通脉冲使集成电路芯片工作，可保证稳定状态下无尖峰脉冲。

3. 修改逻辑设计

如函数 $Y=AB+\bar{B}C$，当 $A=C=1$ 时，与式(4.5.1)吻合，存在竞争-冒险现象。可依照添加项定理添加 AC 项，具体如下：

$$Y=AB+\bar{B}C=AB+\bar{B}C+AC$$

此后，当 $A=C=1$ 时，无论 B 如何变化，输出始终保持高电平，因此，B 的状态变化不会引起竞争-冒险现象。

复习与思考

结合图 4.5.1 谈谈竞争-冒险现象形成的原因。

习题

4-1　填空题

1. 用文字、符号或数字表示特定对象的过程称为________。执行________功能的电路统称为________。3 位的二进制________输出为________线，输入为________线，常形象地将 3 位二进制________描述为________。

2. A 有效，$Y=A$，B 有效，$Y=B$。当高电平为有效输入时，$Y=$________；当低电平为有效输入时，$Y=$________。

3. ________可以将输入二进制代码的状态翻译成输出信号，以表示原来含义。________输出端哪一条有输出信号，取决于所给出的________。

4. 74LS147 为________优先编码器，它与________优先编码器 74LS148 在原理上并无本质区别，主要区别在于 74LS147 不具有________。

5. 当 3-8 译码器 74LS138 控制端________时，如果将地址端 A_2、A_1、A_0作为 3 个输入的自变量，则 8 个输出端输出的就是 A_2、A_1、A_0 3 个的输入变量的________，即________。因此，也常将二进制译码器称为________。

6. 在多路数据传送过程中，往往需要将多路数据中任意一路信号挑选出来，能实现这种逻辑功能的电路称为________。四选一________中，A_1、A_0为选择控制信号，当 $A_1A_0=$________时，输出信号 $Y=D_1$。

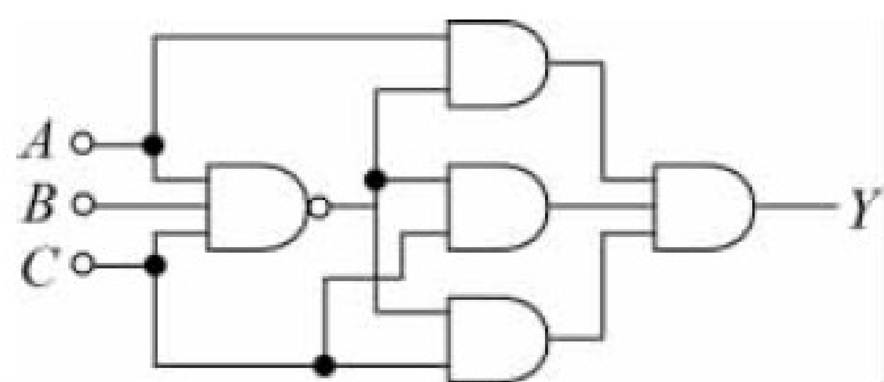
7. 把门电路的两个输入端的信号同时向相反的逻辑电平调变的现象，称为________。把因输入端的竞争而导致输出可能产生不应有的尖峰干扰脉冲（又称过渡干扰脉冲）的现

象称为________。

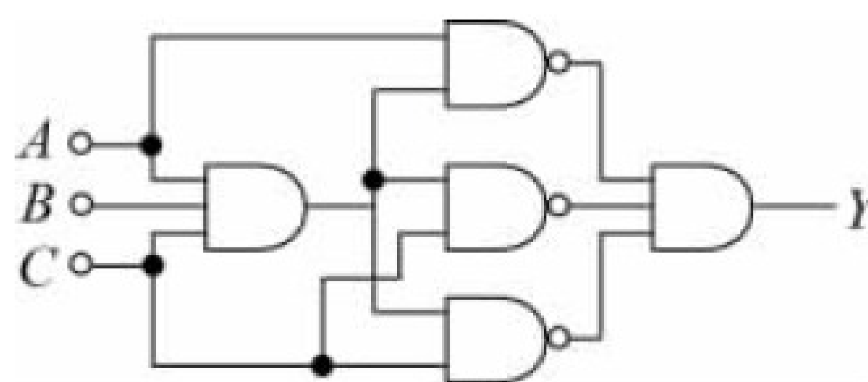
4-2 分析设计题

1. 分析题图 4.1 所示电路的逻辑功能。

2. 分析题图 4.2 所示电路的逻辑功能。



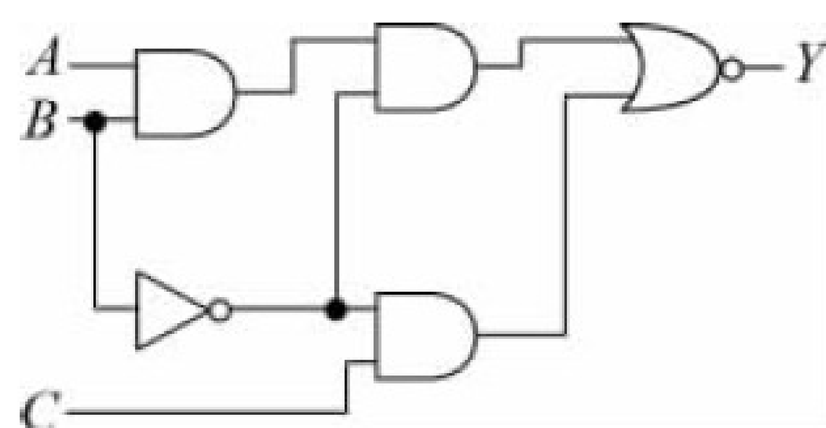
题图 4.1



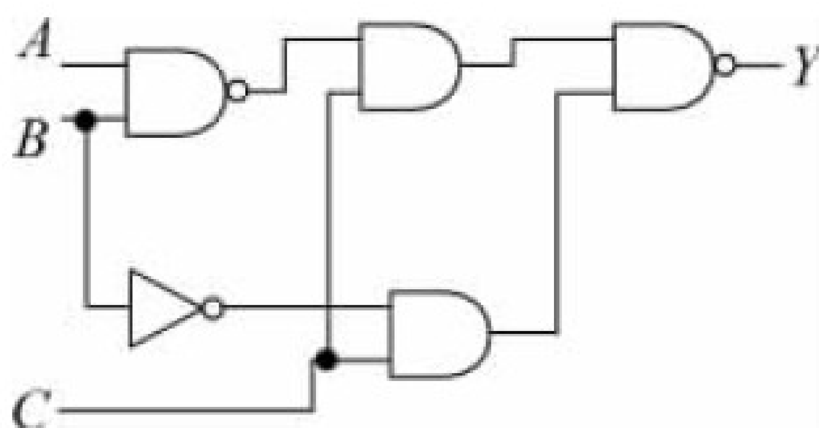
题图 4.2

3. 分析题图 4.3 所示电路的逻辑功能。

4. 分析题图 4.4 所示电路的逻辑功能。



题图 4.3



题图 4.4

5. 试用与非门实现下列函数：

(1) $Y_1(A,B,C)=\sum_m(0,2,5,7)$

(2) $Y_2(A,B,C,D)=\sum_m(0,1,5,7,9,10,12,15)$

(3) $Y_3(A,B,C,D)=\sum_m(1,3,4,6,9,11,12,14)$

(4) $Y_4(A,B,C,D)=\sum_m(0,1,2,4,6,10,14,15)$

(5) $Y_5(A,B,C,D)=\sum_m(1,4,6,7,8,10)+\sum_d(2,5,14)$

(6) $Y_6(A,B,C,D)=\sum_m(1,2,3,7,8)+\sum_d(4,5,9,10,11,14)$

6. 试用或非门实现下列函数：

(1) $Y_1(A,B,C)=\sum_m(0,2,5,7)$

(2) $Y_2(A,B,C,D)=\sum_m(0,1,5,7,9,10,12,15)$

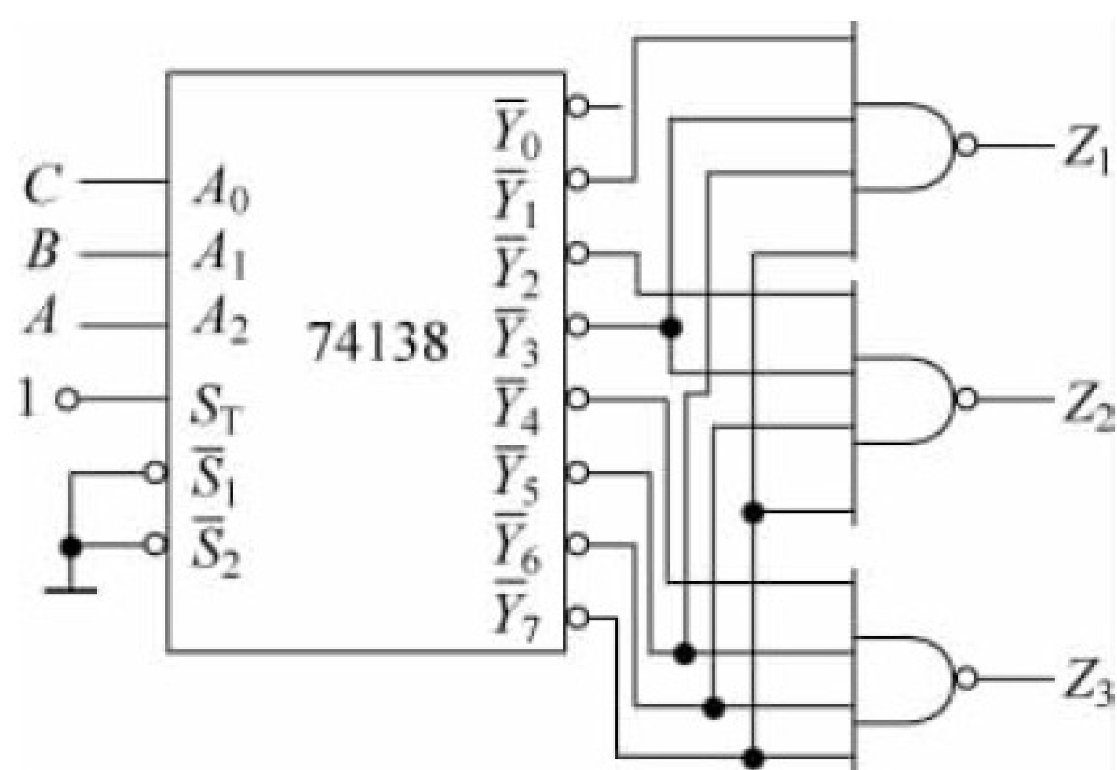
7. 试用 4 个 8-3 编码器 74LS148 结合门电路实现两个 1 个 32-5 优先编码器。

8. 分析题图 4.5 所示电路的逻辑功能。

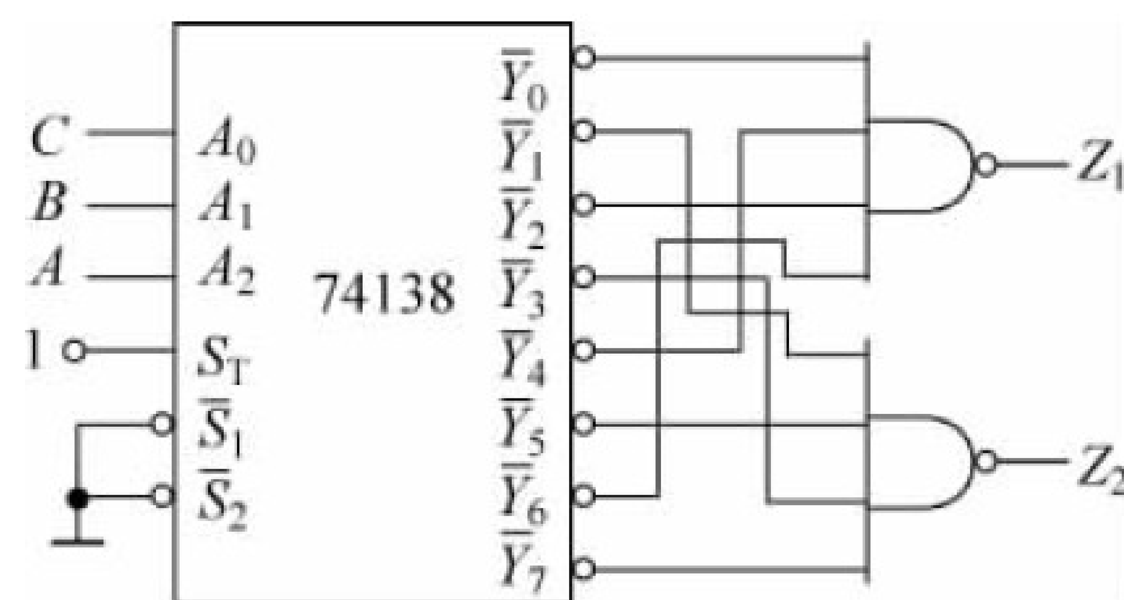
9. 分析题图 4.6 所示电路的逻辑功能。

10. 分析题图 4.7 所示电路的逻辑功能。

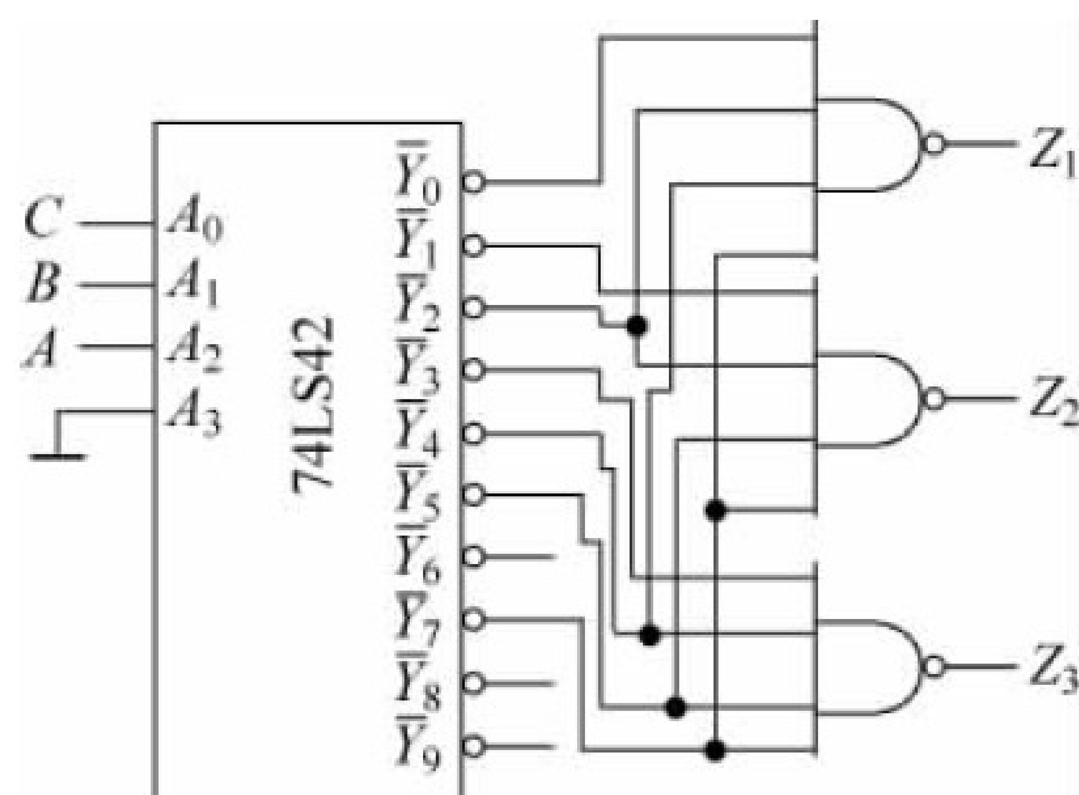
11. 分析题图 4.8 所示电路的逻辑功能。



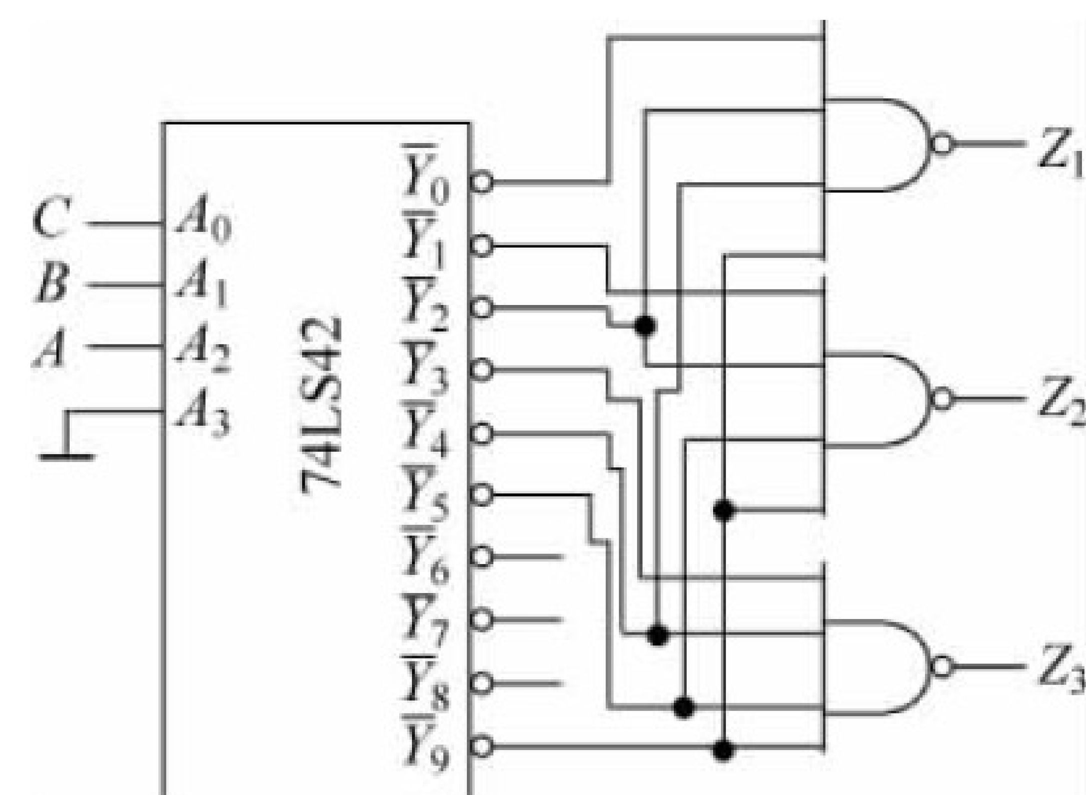
题图 4.5



题图 4.6



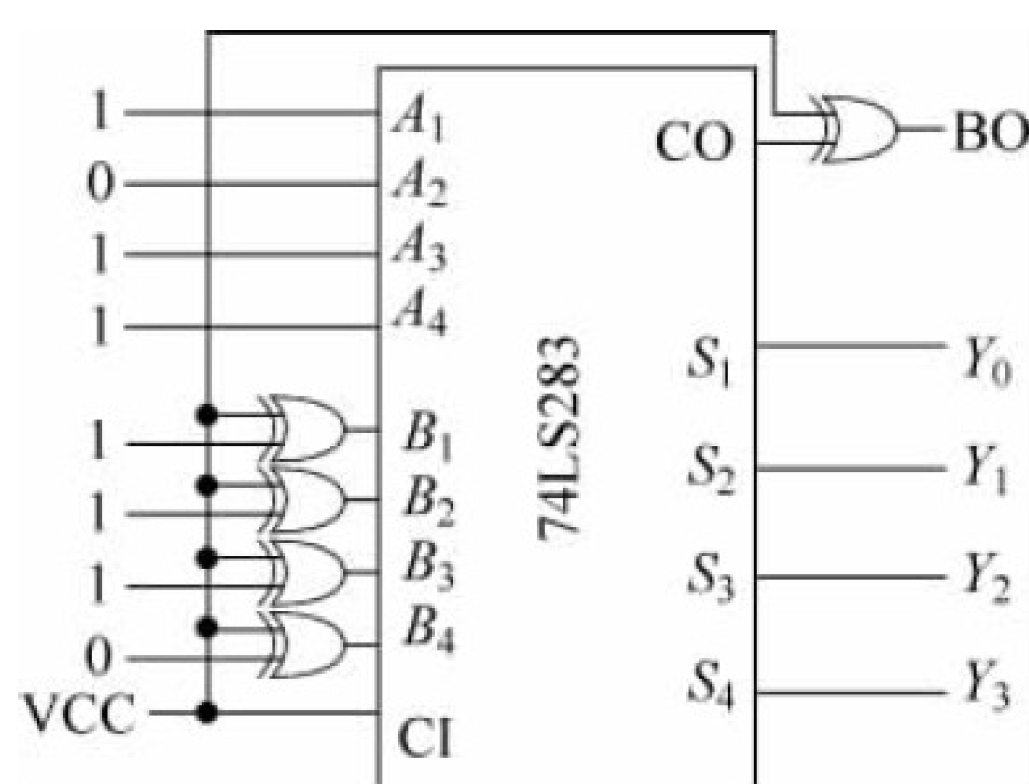
题图 4.7



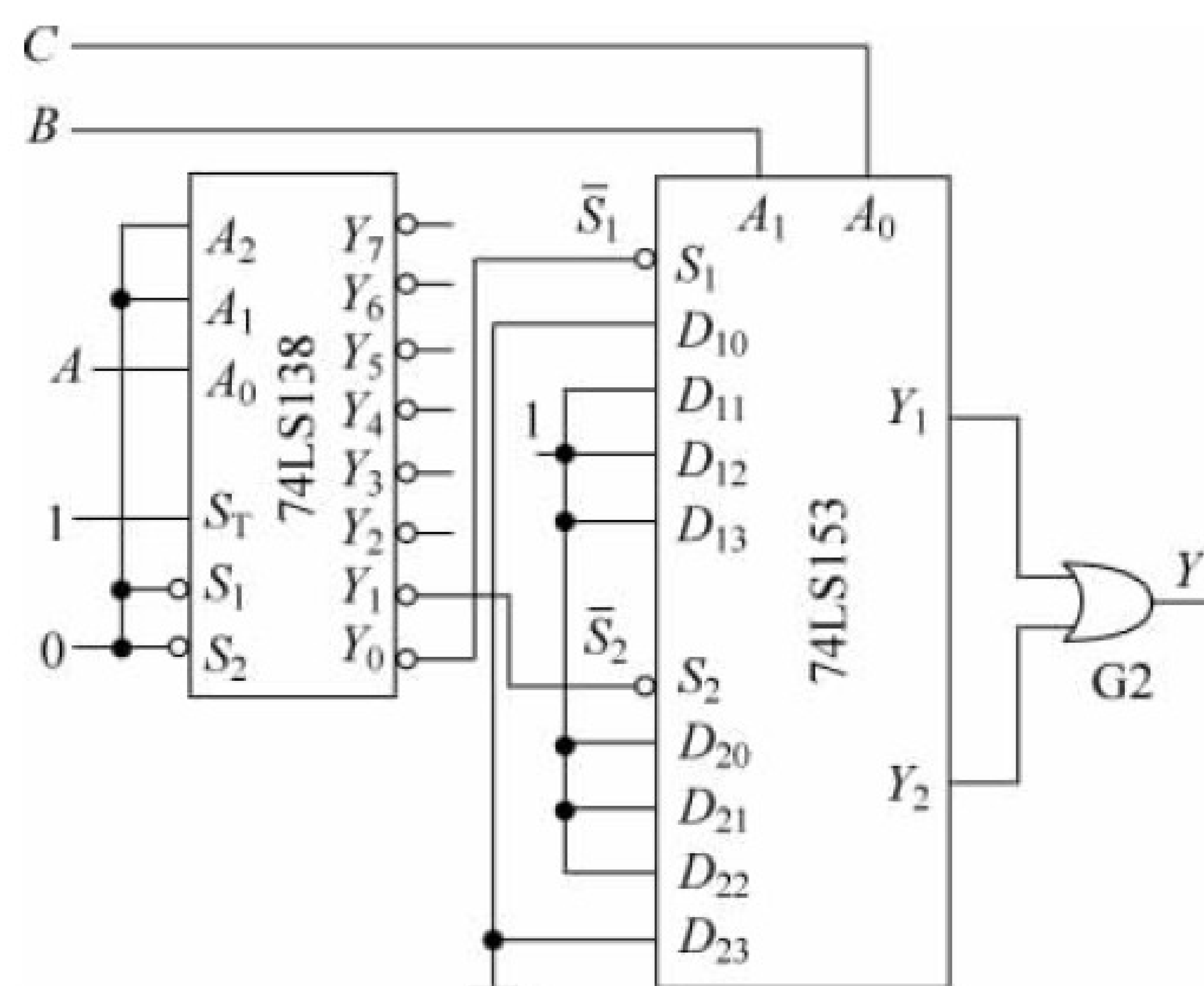
题图 4.8

12. 分析如题图 4.9 所示电路输出的逻辑状态。

13. 分析题图 4.10 所示电路的逻辑功能。



题图 4.9



题图 4.10

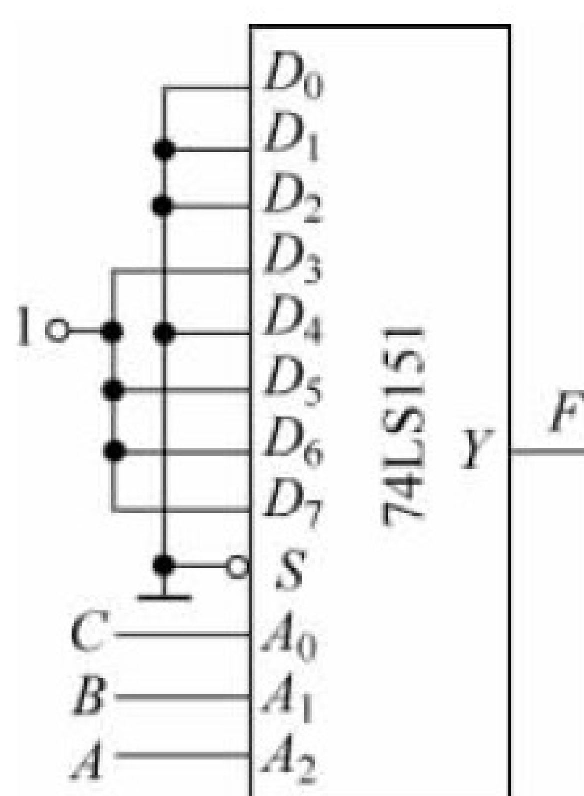
14. 用 74LS283 实现将余 3 码转换为 8421BCD 码的转换电路。

15. 分析题图 4.11 所示电路的逻辑功能。

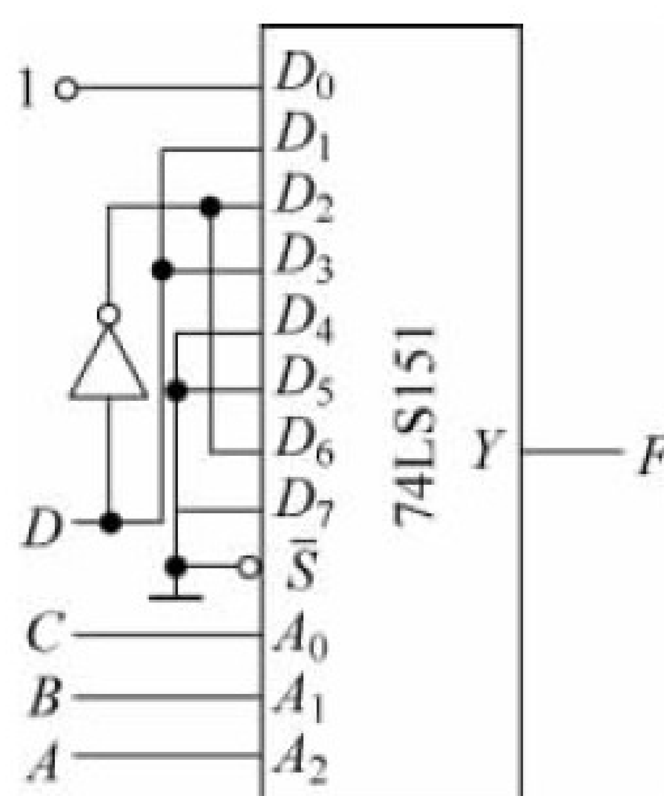
16. 分析题图 4.12 所示电路的逻辑功能。

17. 试用 3-8 译码器 74LS138 实现下列多输出函数的组合逻辑电路。输出的逻辑函数为

$$Z_1 = \overline{A}\overline{B} + ABC$$
$$Z_2 = AB + C$$
$$Z_3 = \overline{A}C + B\overline{C} + A\overline{B}\overline{C} + ABC$$



题图 4.11



题图 4.12

18. 试用八选一数据选择器实现下列函数：

$$Y(A,B,C) = \sum_m (0,2,5,7)$$

$$Y(A,B,C,D) = \sum_m (0,1,5,7,9,10,12,15)$$

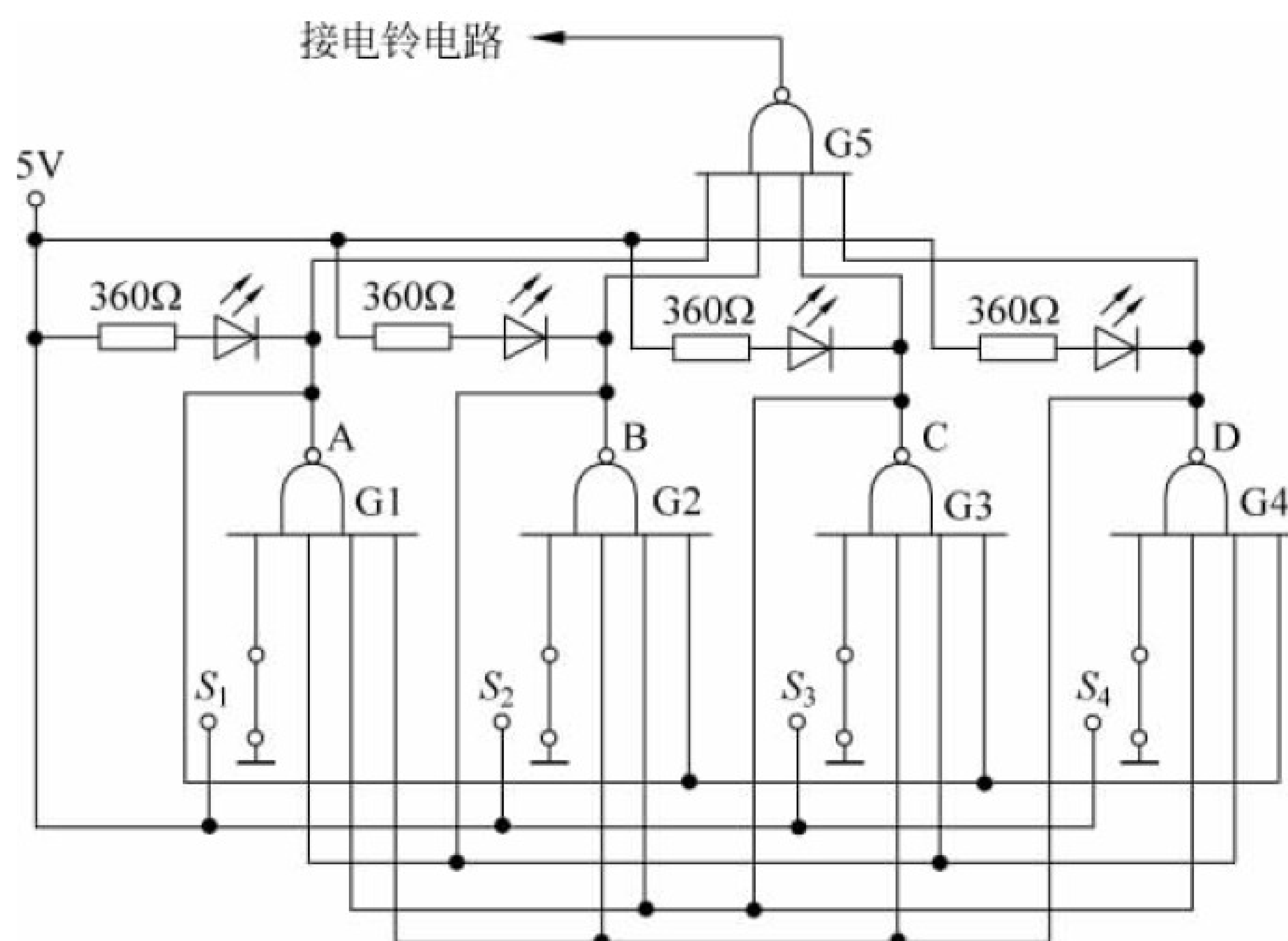
$$Y(A,B,C,D) = \sum_m (1,4,6,7,8,10) + \sum_d (2,5,14)$$

19. 如题图 4.13 所示为智力竞赛抢答电路。在图中，$S_1 \sim S_4$为抢答开关，可以供 4 个参赛组用，发光二极管供各组显示用，当与非门 G5 输出 Y 为高电平时，电铃响。试问：

(1) 当抢答开关在图示位置时，各指示灯能否发亮？电铃能否响？

(2) 当将开关 S_1扳到高电平 6V 时，A 组情况如何？此后再扳动其他组的抢答开关是否起作用？

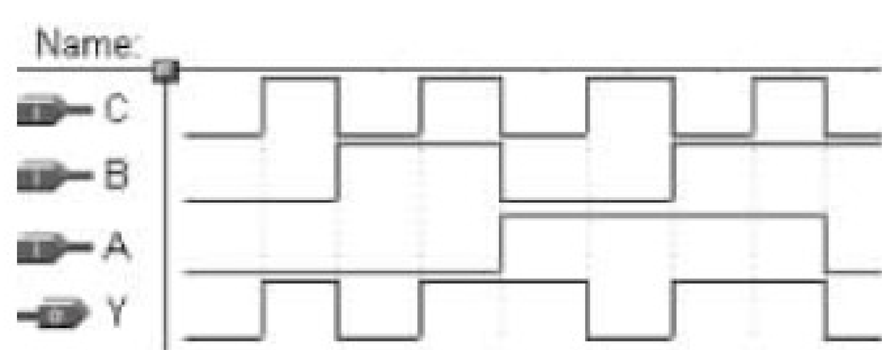
(3) 试画出接在与非门 G5 输出端的电铃电路。



题图 4.13

20. 试用与非门设计一个四变量的多数表决电路。当输入变量 A、B、C、D 有 3 个或 3 个以上为 1 时输出为 1；否则，输出为 0。

21. 已知输入信号 A、B、C 的波形如题图 4.14 所示，试用与非门电路实现输出 Y 波形的组合逻辑电路。



题图 4.14

22. 试用 74LS151 设计一个 2 个 2 位二进制数 A_2A_1、B_2B_1的比较电路。

23. 试用一个四选一数据选择器实现 1 位二进制的全加运算。

24. 试用两片集成 4 位超前进位加法器 74LS283 和必要的门电路设计一个 2 个 4 位二进制数的加/减运算电路，要求控制信号 $M=0$ 时做加法运算，$M=1$ 时做减法运算。

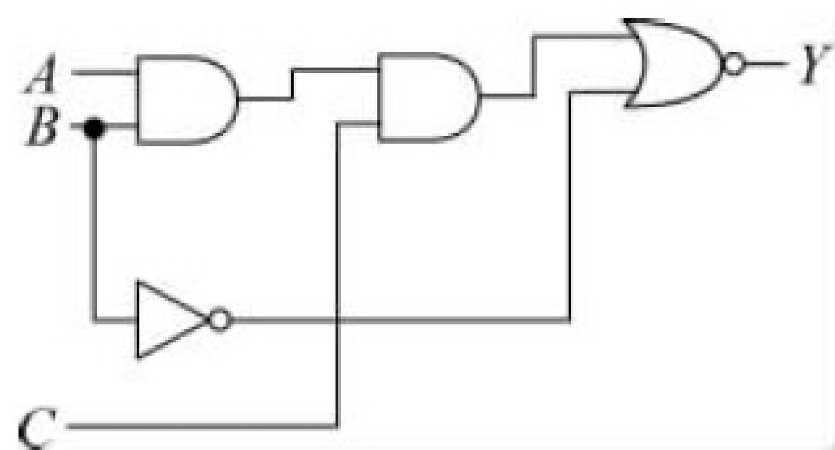
25. 试用两片集成 4 位超前进位加法器 74LS283 和必要的门电路设计实现一个二—十进制加法运算电路(提示：两数相加结果为 9 及以下，二—十进制加法结果和二进制加法结果相同；两数相加结果为 9 以上时，二—十进制加法结果为二进制加法结果基础上加 6)。

26. 试用 74LS85 构成一个 10 位数值比较器。

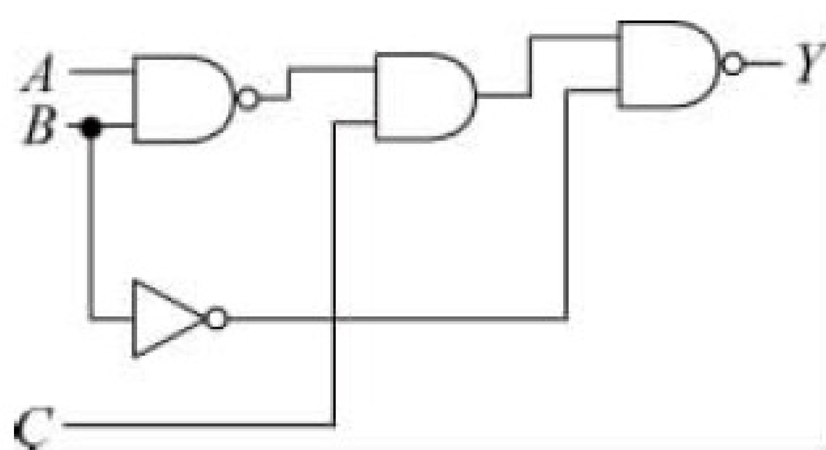
27. 试用两片 74LS85 构成一个“3 个 4 位二进制数的判断电路”。要求能判断 3 个 4 位的二进制数 A、B、C“是否相等、A 是否最大、A 是否最小”，分别给出“3 个数相等、A 最大、A 最小”的输出信号。

28. 试判断如题图 4.15 所示电路是否存在竞争-冒险现象，已知任何瞬间，输入变量只可能有一个改变状态。

29. 试判断如题图 4.16 所示电路是否存在竞争-冒险现象，已知任何瞬间，输入变量只可能有一个改变状态。



题图 4.15



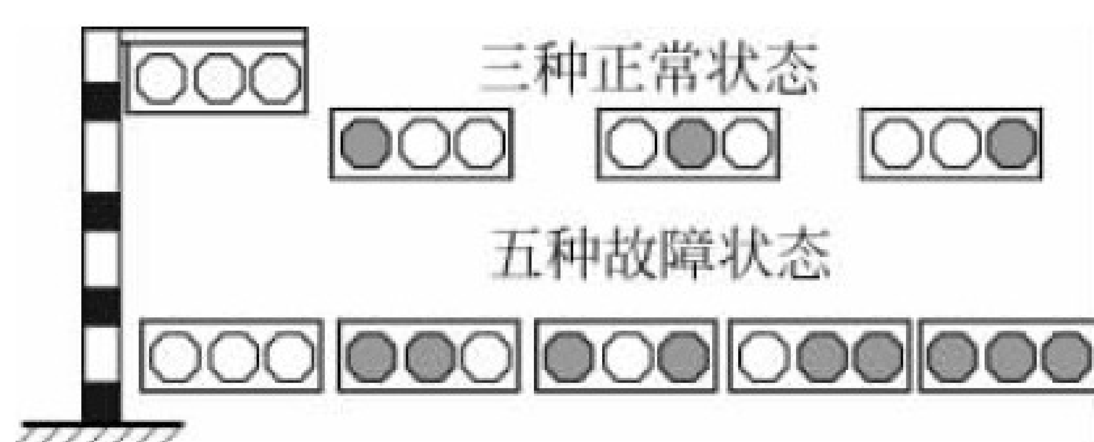
题图 4.16

4-3 应用题

1. 试设计一个监视交通信号灯工作状态的逻辑电路。每组信号灯由红、黄、绿三盏灯组成，如题图 4.17 所示。正常工作情况下，任何时刻必有一盏灯点亮，而且只允许有一盏灯点亮。而当出现其他五种点亮状态时，电路发生故障，这时要求发生故障信号，以提醒维护人员前去修理。

2. 某厂电视机产品有 A、B、C 和 D 4 项质量指标，规定 A 是必须满足要求的，其他三项中只要有任意两项能满足要求，电视机就算合格。试设计该电路。

3. 某实验室有红、黄两个故障指示灯，用来表示三台设备的工作情况；当只有一台设



题图 4.17

备出现故障时，黄灯亮；若有两台设备同时出现故障时，红灯亮；只有当三台设备都出现故障时，才会使红灯和黄灯都亮。试设计一个控制灯亮的电路。

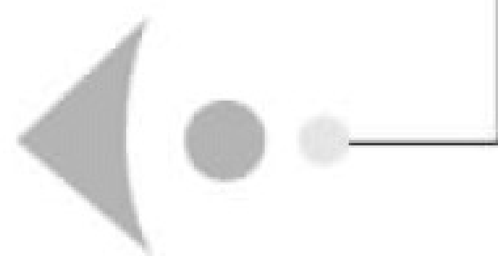
4. 火车旅客列车，分为特快、直快、普快和慢车，并依次为优先通行次序。某火车站对上述列车，规定在同一时间只能有一趟客车从车站开出，即只能给出一个开车信号，试设计出满足上述要求的逻辑电路。

5. 某工厂有三个车间和一个自备电站，站中有两台发电机组 X 和 Y，Y 的发电能力是 X 的 2 倍。如果一个车间开工，则启动 X 机组可满足要求；如果两个车间开工，则启动 Y 机组就能满足要求；如果三个车间同时开工，则两台机组必须全部启动才行。试设计一个控制机组 X 和 Y 启动的电路。

6. 某医院有 1、2、3、4、5、6 号病房 6 间，装有 6 个呼叫器，对应的护士室有 1、2、3、4、5、6 号 6 个指示灯。优先级按照 1、2、3、4、5、6 顺序降低设置，设计该控制电路。



第5章



触发器



本章要点：

本章为本书时序逻辑电路单元基础章。学习本章应理解现态、次态、状态转移真值表、特征方程、状态图、激励表等触发器逻辑功能描述的基础概念；理解 RS、JK、D、T 等常用触发器的逻辑功能特点及其相互转换方法；牢记常用触发器的特征方程；懂得触发器逻辑功能和动作特点是两个不同的概念；理解常见结构触发器动作特点；了解触发器的动态特性。



5.1 什么是触发器

组成组合电路的基本单元电路是门电路，组成时序电路的基本单元电路是触发器。能够存储一位二值(逻辑 0 和逻辑 1)信号的基本单元电路统称为触发器。

为了实现一位二值(逻辑 0 和逻辑 1)信号的存储，触发器应具有两个稳定状态：0 状态和 1 状态(分别对应逻辑 0 和逻辑 1)。当然，为了实现不同状态的存储功能，触发器还应具有保持和修改功能。可保持触发器的 0 状态(或 1 状态)不变，也可将原先保持的 0 状态(或 1 状态)修改为 1 状态(或 0 状态)。具有这样逻辑功能的电路通称为触发器。

可通过如图 5.1.1 所示电路从以下几个方面来理解触发器的概念。

1. 两个稳定状态

如图 5.1.1(a)所示电路由两个或非门首尾交叉连接组成，具有两个输出端(Q 端、$\bar{Q}$ 端)和两个输入端(R、S)。

规定：$Q=0$，$\bar{Q}=1$，为触发器的 0 状态；$Q=1$，$\bar{Q}=0$ 为触发器的 1 状态。进一步分析电路，不难看出，在一定的输入条件下，这两种状态均可成为稳定状态。

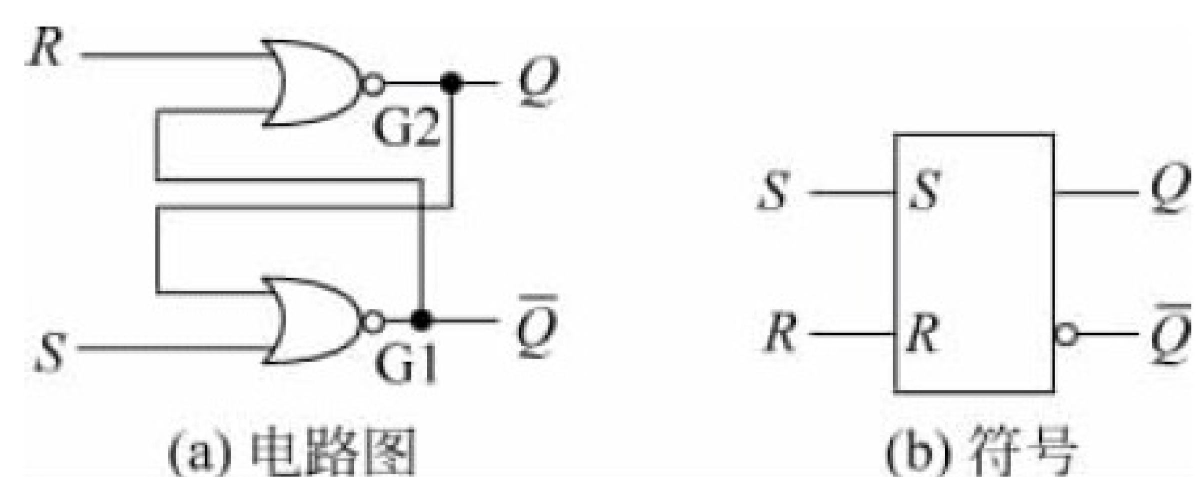


图 5.1.1 用或非门构成的基本 RS 触发器

2. 状态的保持

如果规定高电平为有效信号，当输入信号无效时($R=0$、$S=0$)，触发器保持原来状态不变。分析如下：

若触发器的原来状态为 0 状态，即 $Q=0$，$\overline{Q}=1$。由于 $\overline{Q}=1$ 送到了或非门 G2 的输入端，输出 $Q=0$；而 $Q=0$ 和 $S=0$ 使或非门 G1 导通，维持 $\overline{Q}=1$，即触发器保持原来状态 0 状态。

若触发器的原来状态为 1 状态，即 $Q=1$，$\overline{Q}=0$。由于 $Q=1$ 送到了或非门 G1 的输入端，$\overline{Q}=0$；而 $\overline{Q}=0$ 和 $R=0$ 使或非门 G2 导通，维持 $Q=1$，即触发器保持原来状态 1 状态。

触发器的这种功能称为保持功能。

3. 状态的设置

当输入信号 R 有效时($R=1$，$S=0$)，触发器将变成 0 状态，即 $Q=0$，$\overline{Q}=1$，这种功能称为置 0 功能。

因为当 $R=1$，$S=0$ 时，如果触发器原来是处在 0 状态，则仍保持 0 状态不变，即 $Q=0$，$\overline{Q}=1$ 的状态不会改变；如果触发器原来是处在 1 状态，则由于 $R=1$ 送到了或非门 G2 的输入端，输出 $Q=0$；而 $Q=0$ 和 $S=0$ 使或非门 G1 导通，输出 $\overline{Q}=1$，即触发器为 0 态。

当输入信号 S 有效时($R=0$、$S=1$)，触发器将变成 1 状态，即 $Q=1$，$\overline{Q}=0$，这种功能称为置 1 功能。

因为当 $R=0$，$S=1$ 时，如果触发器原来是处在 1 状态，则仍保持 1 状态不变，即 $Q=1$，$\overline{Q}=0$ 的状态不会改变，如果触发器原来是处在 0 状态，则由于 $S=1$ 送到了或非门 G1 的输入端，输出 $\overline{Q}=0$，而 $\overline{Q}=0$ 和 $R=0$ 使或非门 G2 导通，输出 $Q=1$，即触发器为 1 态。

保持、置 0、置 1 是触发器实现存储功能的基本要求。图 5.1.1(a)所示电路具有保持、置 0、置 1 功能(R 端为置 0 端，S 端为置 1 端)，是组成其他触发器的基础，称为基本 RS 触发器，逻辑符号如图 5.1.1(b)所示。

必须指出的是，当 $R=S=1$ 时，$Q=0$，$\overline{Q}=0$，即不是 0 状态，也不是 1 状态，没有意义，应避免。此外，当 R、S 的高电平同时撤除时，此时，$R=S=0$，$Q=\overline{Q}=0$，为非稳定状态。若 G2 首先跳变到 1，输出 $Q=1$，反馈到 G1 输入端，锁定 $\overline{Q}=0$，触发器进入触发器 1 状态。类似地，若 G1 首先跳变到 1，输出 $\overline{Q}=1$，反馈到 G2 输入端，锁定 $Q=0$，触发器进入触发器 0 状态。

可见，当 R、S 的高电平同时撤除时，触发器的状态不能确定，可能是 0 状态，也可能是 1 状态。

当然，基本 RS 触发器还有其他形式，如图 5.1.2 所示电路为用与非门组成的基本 RS 触发器。

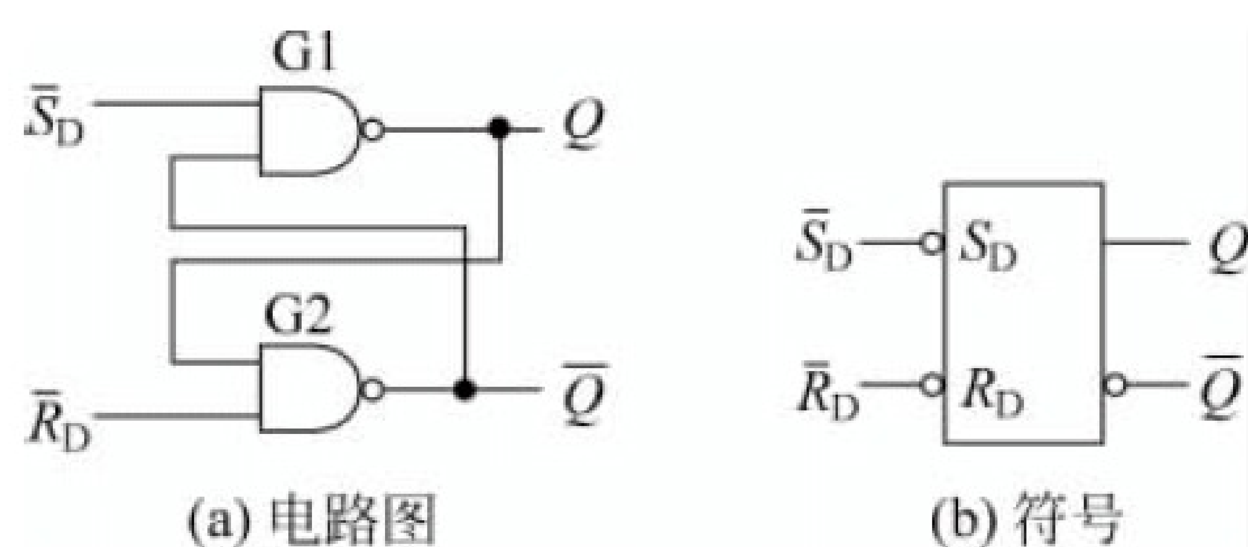


图 5.1.2 用与非门构成的基本 RS 触发器

如图 5.1.2(a)所示电路由两个与非门首尾交叉连接组成，具有两个输出端（Q 端、$\bar{Q}$ 端）和两个输入端（$\bar{R}_D$、$\bar{S}_D$）。类似地，$Q=0$，$\bar{Q}=1$，为触发器的 0 状态，$Q=1$，$\bar{Q}=0$ 为触发器的 1 状态。进一步分析电路，不难看出，在一定的输入条件下，这两种状态均可成为稳定状态。

如果规定低电平为有效信号，则当输入信号无效时（$\bar{R}_D=1$，$\bar{S}_D=1$），触发器保持原来状态不变。分析如下：

若触发器的原来状态为 0 状态，即 $Q=0$，$\bar{Q}=1$。由于 $Q=0$ 送到了与非门 G2 的输入端，输出 $\bar{Q}=1$；而 $\bar{Q}=1$ 和 $\bar{S}_D=1$ 使与非门 G1 导通，维持 $Q=0$，即触发器保持原来状态 0 状态。

若触发器的原来状态为 1 状态，即 $Q=1$，$\bar{Q}=0$。由于 $\bar{Q}=0$ 送到了与非门 G1 的输入端，输出 $Q=1$，而 $Q=1$ 和 $\bar{R}_D=1$ 使与非门 G2 导通，维持 $\bar{Q}=0$，即触发器保持原来状态 1 状态。

当输入信号 $\bar{R}_D$ 有效时（$\bar{R}_D=0$，$\bar{S}_D=1$），触发器将变成 0 状态，即 $Q=0$，$\bar{Q}=1$，具有置 0 功能。当 $\bar{R}_D=0$，$\bar{S}_D=1$ 时，如果触发器原来处在 0 状态，则仍保持 0 状态不变，即 $Q=0$，$\bar{Q}=1$ 的状态不会改变；如果触发器原来处在 1 状态，则由于 $\bar{R}_D=0$ 送到了与非门 G2 的输入端，输出 $\bar{Q}=1$；而 $\bar{Q}=1$ 和 $\bar{S}_D=1$ 使与非门 G1 导通，输出 $Q=0$，即触发器为 0 状态。

当输入信号 $\bar{S}_D$ 有效时（$\bar{R}_D=1$，$\bar{S}_D=0$），触发器将变成 1 状态，即 $Q=1$，$\bar{Q}=0$，具有置 1 功能。当 $\bar{R}_D=1$，$\bar{S}_D=0$ 时，如果触发器原来处在 1 状态，则仍保持 1 状态不变，即 $Q=1$，$\bar{Q}=0$ 的状态不会改变；如果触发器原来处在 0 状态，则由于 $\bar{S}_D=0$ 送到了与非门 G1 的输入端，输出 $Q=1$，而 $Q=1$ 和 $\bar{R}_D=1$ 使与非门 G2 导通，输出 $\bar{Q}=0$，即触发器为 1 状态。

可见，如图 5.1.2(a)所示电路具有保持、置 0、置 1 等功能，是另一种形式的基本 RS 触发器，是组成其他触发器的基础，逻辑符号如图 5.1.2(b)所示。

复习与思考

试分析如图 5.1.2(a)所示触发器哪种输入组合没有意义，应尽量避免。



5.2 基本 RS 触发器的逻辑功能描述

如图 5.1.1(a)、图 5.1.2(a)所示电路为基本 RS 触发器，是组成其他触发器的基础。本节以如图 5.1.1(a)所示基本 RS 触发器为例，从以下几个方面介绍其逻辑功能的描述方法。

1. 现态与次态

触发器在输入信号作用之前所处的原稳定状态称为现态，用 Q^n 和 $\bar{Q}^n$ 表示(为书写方便，上标 n 也可以不写)。触发器在输入信号作用下所处的新的状态称为次态，用 Q^{n+1} 和 $\bar{Q}^{n+1}$ 表示。

显然，Q^{n+1}、Q^n 的值由两根输出线的状态决定。若当前输出 $Q=1$，$\bar{Q}=0$，则 $Q^n=0$。改变输入 R、S，触发器将要进入的新的状态记为 Q^{n+1}。

2. 状态转移真值表

将 Q^n、R、S 视为自变量，Q^{n+1} 视为函数。根据工作原理的分析，可以列出如表 5.2.1 所示的状态转移真值表。

表 5.2.1 状态转移真值表

R	S	Q^n	Q^{n+1}
0	0	0	0
0	1	0	1
1	0	0	0
1	1	0	×
0	0	1	1
0	1	1	1
1	0	1	0
1	1	1	×

由状态转移真值表的第 1、5 行可以看出，当 $R=0$、$S=0$ 时，$Q^{n+1}=Q^n$。触发器保持现态不变。

由状态转移真值表的第 2、6 行可以看出，S 为置 1 输入端，当 $R=0$、$S=1$ 时，$Q^{n+1}=1$。

类似地，状态转移真值表的第 3、7 行体现 R 为置 0 输入端，当 $R=1$、$S=0$ 时，$Q^{n+1}=0$。

5.1 节指出，当 $R=S=1$ 时，输出 $Q=0$，$\bar{Q}=0$，既不是 0 状态，也不是 1 状态，没有意义，应避免。逻辑代数中，用无关项来描述这种逻辑关系，用×表示(第 4、8 行)。

无关项的含义在逻辑运算基础相关内容中做过详细介绍，在实际使用中，应防止 $R=S=1$ 这种情况的出现，应对触发器的输入端加以约束限制。显然，若 R、S 满足 $RS=0$，则能保证输入端不会同时出现高电平。

必须指出，上面的无关项是由电路的结构决定的。因为不允许出现 $R=S=1$ 的情况，

所以 $R=S=1$ 的输入组合与电路的输出无关，常用功能不定描述 $R=S=1$ 的输入组合对应的输出状态。

主要有两重含义：

(1) 当 $R=S=1$ 时，输出 $Q=0,\bar{Q}=0$。对触发器来讲，$Q=0$、$\bar{Q}=0$ 这种状态毫无意义，因为这既不是触发器的 0 状态，也不是触发器的 1 状态。

(2) 当 R、S 端的高电平同时撤除(即由全 1 同时跳变为全 0)，由于或非门 G1、G2 的平均传输时间的离散性及外部干扰信号的影响，使得触发器的状态无法确定(既可能是 0 状态，也可能是 1 状态)。

3. 特征方程

描述触发器逻辑功能的函数表达式称为特征方程，或者称为特性方程。由表 5.2.1 经过如图 5.2.1 所示的卡诺图化简，可得

$$\begin{cases} Q^{n+1} = S + \bar{R}Q^{n} \\ RS = 0 \end{cases} \tag{5.2.1}$$

Q^n \ RS	00	01	11	10
0		1	×	
1	1	1	×	

图 5.2.1 基本触发器卡诺图

式中：$RS=0$ 为约束条件(不允许输入端 R、S 同时为 1)。

特征方程能完整描述触发器的逻辑功能，是描述触发器逻辑功能的又一基本方法。

状态转移真值表、特征方程类似组合逻辑电路的真值表、逻辑函数表达式。因为触发器输出信号不仅取决于该时刻电路的输入信号，而且决定于电路原来的状态，而状态转移真值表、特征方程虽然可以描述触发器的逻辑功能，却不能直观反映电路状态在输入激励下的变化，因此，触发器还经常采用状态转移图(简称状态图)、时序图、激励表等描述手段。

4. 状态图

触发器的逻辑功能还可以采用图形的方式来描述，即状态转移图(简称状态图)。

如图 5.2.2 所示为 RS 触发器的状态转移图。图中两个小圆分别代表触发器的两个稳定状态，箭头表示在输入信号作用下状态转移的方向，箭头旁的标注表示转移时的条件。

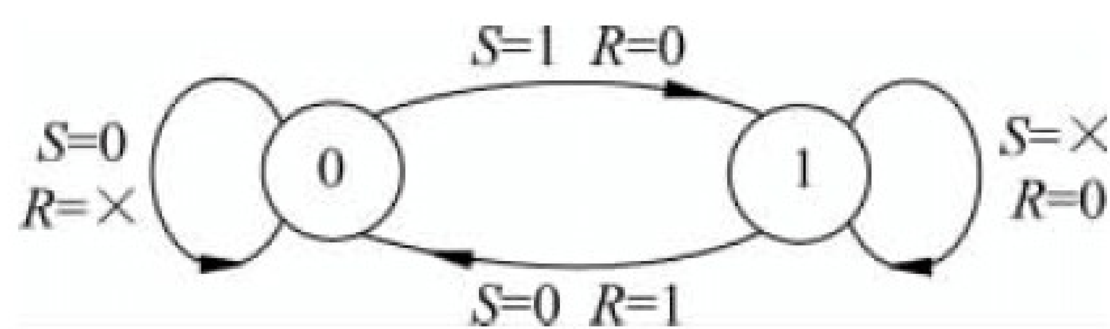


图 5.2.2 RS 触发器状态图

状态 0 旁的标注表示状态 0 保持不变的条件为 $R=\times$，$S=0$(真值表第 1、3 行)。类似地，状态 1 保持不变的条件为 $R=0$，$S=\times$(真值表第 5、6 行)。

进一步分析状态图，如果触发器当前稳定状态(现态)是 $Q^n=0$，则在输入信号 $S=1$、$R=0$ 的条件下，触发器转移至下一稳定状态(次态)$Q^{n+1}=1$(真值表第 7 行)。

类似地，如果触发器的原状态稳定为 $Q^n=1$，则在输入信号 $S=0$，$R=1$ 的条件下，触发器转移至下一稳定状态 $Q^{n+1}=0$(真值表第 2 行)。

可见，如图 5.2.2 所示的状态图与表 5.2.1、式(5.2.1)描述的逻辑功能是一致的，只不

过状态图更直观地反映了触发器的状态变化特点。

既然三者描述的逻辑功能是一致的，当然也就可以相互求解。可由状态图求出状态转移真值表、特征方程；也可以由状态转移真值表、特征方程画出状态图。

由式(5.2.1)求 RS 触发器状态图的方法：先画两个小圆(其中的一个圆内填 0，另一个填 1)，分别代表触发器的两个稳定状态。接着画出所有可能的状态转移。由式(5.2.1)(或表 5.2.1)求出每一个可能的状态转移所应具有的输入条件并标注在旁边，即可得到如图 5.2.2 所示的 RS 触发器的状态图。

5. 激励表

由图 5.2.2 可以很方便地列出表 5.2.2。表 5.2.2 表示了触发器由原状态 Q^n 转移至确定要求的下一新状态 Q^{n+1} 时，对输入信号的要求。所以，表 5.2.2 称为触发器的激励表或称为驱动表。

表 5.2.2　激励表

Q^n	Q^{n+1}	R	S
0	0	×	0
0	1	0	1
1	0	1	0
1	1	0	×

显然，触发器的激励表更直观地反映了触发器每一个可能的状态转移所应具有的输入条件。

关于触发器逻辑功能的时序图描述方法，将在 5.4 节介绍。

复习与思考

5.2.1　试总结触发器的状态转移真值表与组合逻辑电路真值表的异同。

5.2.2　试总结触发器逻辑功能的描述方法与组合逻辑电路逻辑功能描述方法的异同。

5.3　常见触发器的逻辑功能

触发器是构成时序电路的基本单元电路。按照触发器逻辑功能的不同，触发器又可分为 RS 功能触发器、JK 功能触发器、D 功能触发器、T 功能触发器等。

1. RS 触发器



时序电路的工作信号为时钟信号[①]，其逻辑功能由其状态转移真值表决定。

① 时钟信号是时序逻辑电路的工作信号。如奔腾Ⅳ 3GHz CPU 中的 3GHz 便是指时钟信号的工作频率。在本书中，时序电路一般使用时钟信号仿真。在 5.4 节，为帮助读者理解、便于仿真实现，在有些场合下没有采用时钟信号。

在时钟信号作用下逻辑功能符合表 5.2.1 所规定的逻辑功能的触发器称为 RS 触发器，其逻辑符号见表 5.3.5。RS 触发器的特征方程，状态图、激励表见 5.2 节。

RS 触发器具有保持、置 0、置 1 三种功能，应用时应遵循 $RS=0$ 的输入约束。RS 触发器逻辑功能的主要特点如下：

$R=1, S=0, Q^{n+1}=0$ （R：Reset 置 0）

$R=0, S=1, Q^{n+1}=1$ （S：Set 置 1）

$R=0, S=0, Q^{n+1}=Q^n$（保持）

不允许 R、S 同时为 1。

2. JK 触发器

在时钟信号作用下逻辑功能符合表 5.3.1 所规定的逻辑功能的触发器称为 JK 触发器。

表 5.3.1 JK 触发器状态转移真值表

Q^n	J	K	Q^{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

真值表第 3、7 行表明，J 输入端为置 1 输入端。当 J 输入有效、K 无效时，触发器置 1。真值表第 2、6 行表明，K 输入端为置 0 输入端。当 K 输入有效、J 无效时，触发器置 0。真值表第 1、5 行表明，当 $J=K=0$ 时，触发器状态保持不变。真值表第 4、8 行表明，当 $J=K=1$ 时，触发器状态翻转。

由表 5.3.1 可做出卡诺图，如图 5.3.1 所示，化简卡诺图，可求出 JK 触发器的特征方程：

$$Q^{n+1}=J\bar{Q}^n+\bar{K}Q^n \qquad (5.3.1)$$

Q^n \ JK	00	01	11	10
0			1	1
1	1			1

图 5.3.1 JK 触发器卡诺图

由式(5.3.1)可知：当 $J=0, K=0$ 时，$Q^{n+1}=0\bar{Q}^n+\bar{0}Q^n=Q^n$，具有保持功能；当 $J=0, K=1$ 时，$Q^{n+1}=0\bar{Q}^n+\bar{1}Q^n=0$，具有置 0 功能；当 $J=1, K=0$ 时，$Q^{n+1}=1\bar{Q}^n+\bar{0}Q^n=1$，具有置 1 功能；当 $J=1, K=1$ 时，$Q^{n+1}=1\bar{Q}^n+\bar{1}Q^n=\bar{Q}^n$，具有翻转功能。

对照 RS 触发器和 JK 触发器逻辑功能，有

$$J\rightarrow S,\quad K\rightarrow R$$

此外,JK 触发器还额外具有翻转功能。JK 触发器逻辑功能的主要特点如下:

$J=1, K=0, Q^{n+1}=1$(置 1)

$J=0, K=1, Q^{n+1}=0$(置 0)

$J=0, K=0, Q^{n+1}=Q^n$ (保持)

$J=1, K=1, Q^{n+1}=\bar{Q}^n$(翻转)

由于 JK 触发器具有保持、置 0、置 1、翻转等多种功能,功能比较齐全,所以 JK 触发器也称为全功能触发器。

由表 5.3.1[或式(5.3.1)]可求出如图 5.3.2 所示的 JK 触发器状态图。由状态图可求出 JK 触发器的激励表,如表 5.3.2 所示。

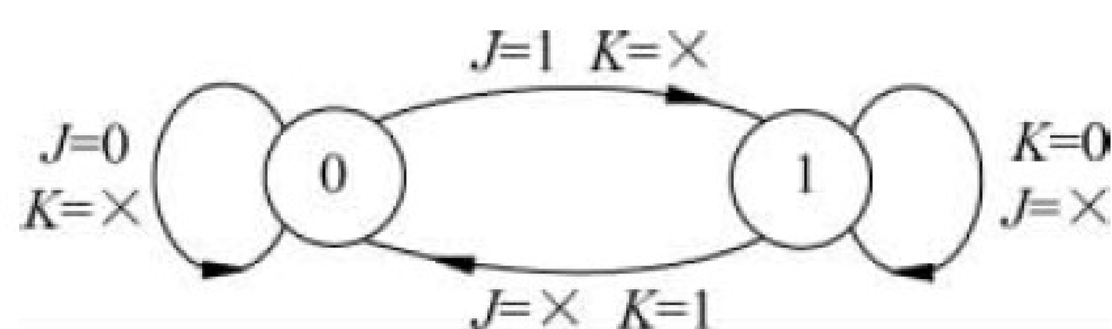


图 5.3.2 JK 触发器状态图

表 5.3.2 JK 触发器激励表

Q^n	Q^{n+1}	J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

可参考 RS 触发器状态图的分析方法分析 JK 触发器状态图。如果触发器当前稳定状态(现态)是 $Q^n=0$,则在输入信号 $J=1, K=\times$ 的条件下,触发器转移至下一稳定状态(次态)$Q^{n+1}=1$(真值表第 3、4 行)。类似分析其余的状态转移条件,可由状态图求出如表 5.3.1 所示的状态转移真值表。



JK 触发器功能齐全,应用时无输入约束,得到了广泛应用。

3. T 触发器

在时钟信号作用下逻辑功能符合表 5.3.3 所规定的逻辑功能的触发器称为 T 触发器。分析如表 5.3.3 所示的真值表,可总结 T 触发器的逻辑功能特点如下:

$T=1, Q^{n+1}=\bar{Q}^n$(翻转,对应真值表第 2、4 行)

$T=0, Q^{n+1}=Q^n$(保持,对应真值表第 1、3 行)

表 5.3.3 T 触发器状态转移真值表

T	Q^n	Q^{n+1}
0	0	0
1	0	1
0	1	1
1	1	0

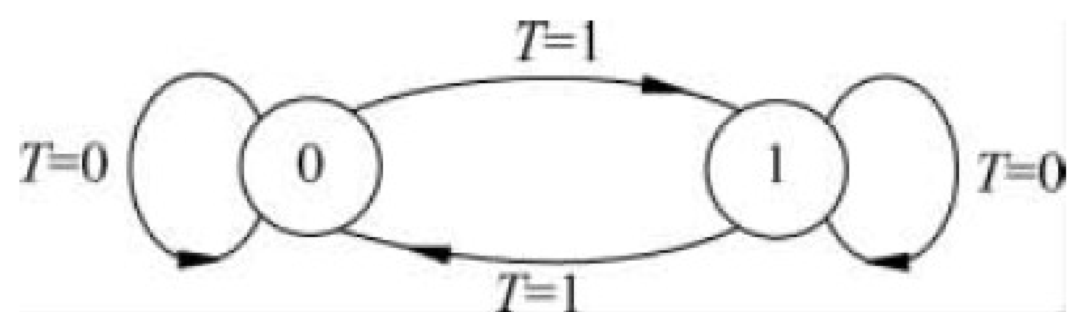


图 5.3.3 T 触发器状态图

可由表 5.3.3 直接写出 T 触发器的特征方程如下:

$$Q^{n+1}=T\bar{Q}^n+\bar{T}Q^n \tag{5.3.2}$$

由表 5.3.3[或式(5.3.2)]可求出如图 5.3.3 所示的 T 触发器状态图。

可参考 RS 触发器、JK 触发器状态图的分析方法分析 T 触发器状态图，进而由状态图求出相应的状态转移真值表。

4. D 触发器

在时钟信号作用下逻辑功能符合表 5.3.4 所规定的逻辑功能的触发器称为 D 触发器。

分析如表 5.3.4 所示的真值表，可总结 D 触发器的逻辑功能特点如下：

$D=0, Q^{n+1}=0$（置 0，对应真值表第 1、2 行）

$D=1, Q^{n+1}=1$（置 1，对应真值表第 3、4 行）

从上面的逻辑特点可以看出，D 触发器本身没有状态保持功能。各类触发器逻辑功能的前提条件是时钟信号的作用。当无时钟信号作用时，触发器将保持状态不变，D 触发器正是利用时钟信号来实现状态的保持的。

可由表 5.3.4 直接写出 D 触发器的特征方程如下：

$$Q^{n+1}=D \tag{5.3.3}$$

由式(5.3.3)可知，当输入信号 $D=1$ 时，$Q^{n+1}=1$；当输入信号 $D=0$ 时，$Q^{n+1}=0$。

由表 5.3.4[或式(5.3.3)]可求出如图 5.3.4 所示的 D 触发器状态图。

表 5.3.4　D 触发器状态转移真值表

D	Q^n	Q^{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

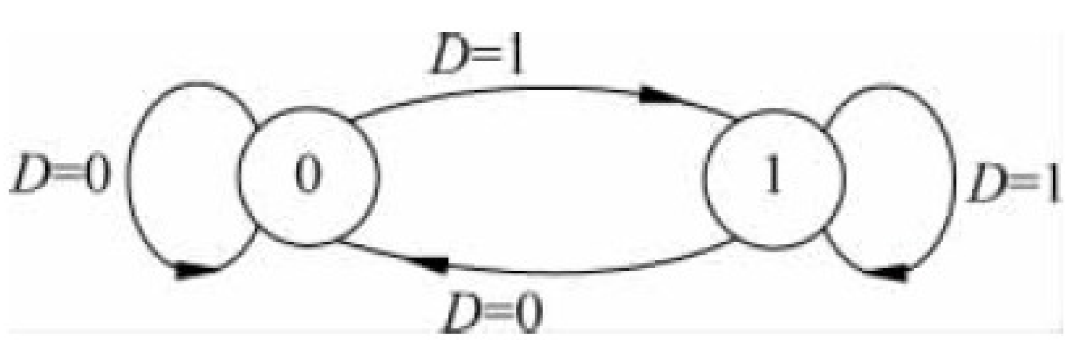


图 5.3.4　D 触发器状态图

D 触发器功能简单，应用时无输入约束，应用十分广泛。

本书中使用的常见触发器逻辑符号及其简单描述如表 5.3.5 所示。其中的涉及触发器电路结构的术语（如上升沿触发），将在 5.4 节介绍。

表 5.3.5　本书中使用的常见触发器逻辑符号及其简单描述

逻 辑 符 号	触发器描述	触 发 方 式	触发器描述
	维持阻塞 RS 触发器（上升沿触发）、同步 RS 触发器 不加说明为维持阻塞 RS 触发器		维持阻塞 JK 触发器（上升沿触发）、同步 JK 触发器 不加说明为维持阻塞 JK 触发器
	主从 RS 触发器、下降沿 RS 触发器		主从 JK 触发器、下降沿 JK 触发器

续表

逻辑符号	触发器描述	触发方式	触发器描述
	维持阻塞D触发器(上升沿触发)、同步D触发器 不加说明为维持阻塞D触发器		主从D触发器、下降沿D触发器

表中触发器的逻辑符号为带异步输入端的逻辑符号，当不用异步功能时也可以不画出。异步的含义为不受上升沿等触发特性的约束，直接引起触发器状态的变化。

常见触发器逻辑功能及其主要特点汇总如表5.3.6所示。

表5.3.6　常见触发器逻辑功能及其主要特点

RS功能触发器	JK功能触发器
特征方程：$Q^{n+1}=S+\bar{R}Q^n$ 主要特点： $R=1,S=0,Q^{n+1}=0$ $R=0,S=1,Q^{n+1}=1$ $R=0,S=0,Q^{n+1}=Q^n$ 不允许R、S同时为1	特征方程：$Q^{n+1}=J\bar{Q}^n+\bar{K}Q^n$ 主要特点： $K=1,J=0,Q^{n+1}=0$ $K=0,J=1,Q^{n+1}=1$ $K=0,J=0,Q^{n+1}=Q^n$ $K=1,J=1,Q^{n+1}=\bar{Q}^n$
D功能触发器	**T功能触发器**
特征方程：$Q^{n+1}=D$ 主要特点： $D=1,Q^{n+1}=1$ $D=0,Q^{n+1}=0$	特征方程：$Q^{n+1}=T\bar{Q}^n+\bar{T}Q^n$ 主要特点： $T=1,Q^{n+1}=\bar{Q}^n$(翻转) $T=0,Q^{n+1}=Q^n$(保持)

必须指出，不同逻辑功能的触发器是可以相互转换的。可用JK功能触发器方便地构成T功能触发器。

对照JK功能触发器、T功能触发器的特征方程，令$J=K=T$，则JK功能触发器的特征方程和T功能触发器的特征方程相同，二者具有相同的逻辑功能，此时，JK功能触发器便成为T功能触发器。



可通过例题进一步理解不同逻辑功能的触发器的相互转换。

【例5.3.1】 请用JK触发器实现D功能触发器。

解 (1) 电路框图。根据题目要求，输入为D，输出为Q、$\bar{Q}$。用JK触发器实现，可画出电路框图，如图5.3.5(a)所示。

(2) 求出转换电路逻辑函数。令JK功能触发器、D功能触发器的特征方程相等，有

$$Q^{n+1}=J\bar{Q}^n+\bar{K}Q^n=D=D\bar{Q}^n+\bar{\bar{D}}Q^n$$

由图5.3.5(a)可知，输入为D，输出为J、K，可求出转换电路逻辑函数

$$J=D,\quad K=\bar{D}$$

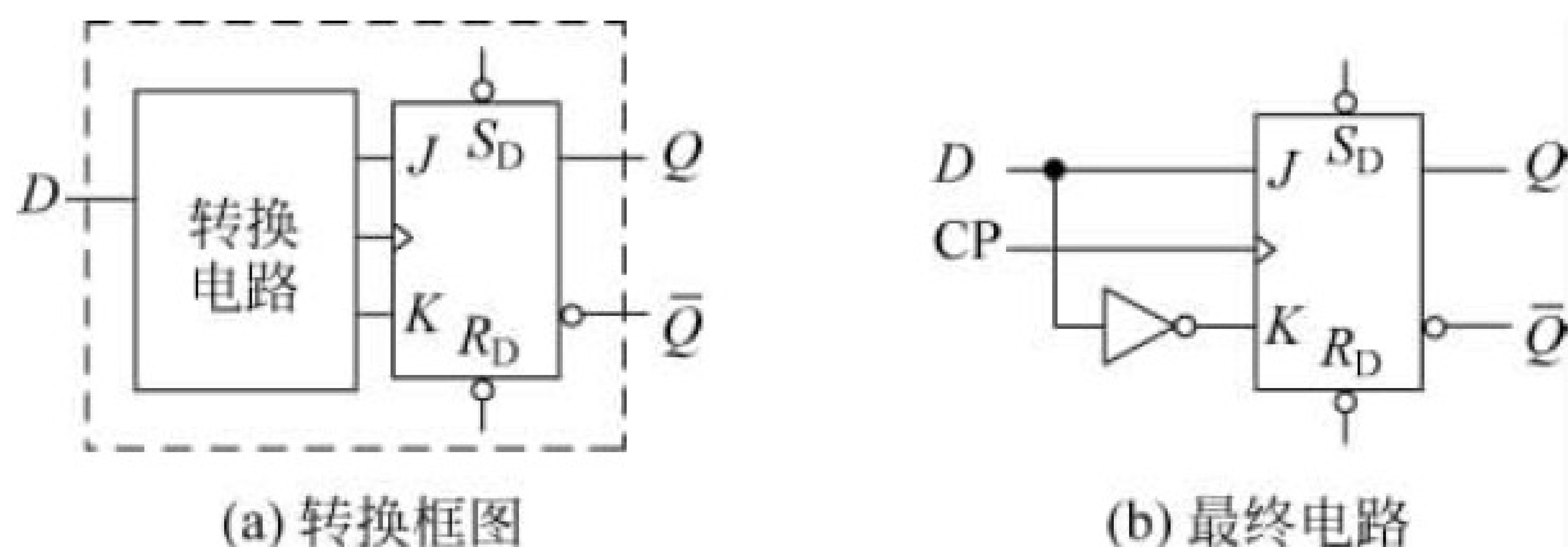


图 5.3.5　例 5.3.1 的图

(3) 画出电路。画出电路如图 5.3.5(b)所示。图中的 JK 触发器为上升沿触发的 JK 触发器,CP 为外部输入时钟信号。

复习与思考

5.3.1　参考例 5.3.1 画出用 D 功能触发器实现 JK 触发器的转换框图并画出转换电路。

5.3.2　T 功能触发器应用十分广泛,市场上却很难直接买到,为什么?

5.4　触发器的动作特点

理解时序逻辑电路的动作特点是时序电路应用的基础,理解触发器的动作特点是理解时序电路动作特点的基础。



5.4.1　动作特点的引入

绝大多数读者都有按门铃的经验。显然,门铃电路具有一个输入(门铃按钮)、一个输出(喇叭)。当人们以某种方式按门铃按钮并被系统接受时,门铃的喇叭将发声。显然,门铃的发声过程涉及两个问题:门铃如何按才能被系统确认;门铃的喇叭发出什么声音。

触发器具有两种状态,其状态转移需要特定的输入条件。把被触发器确认的外部输入当作“触”(类似门铃如何按才能被系统确认),触发器被“触”后,将产生状态转移,这便是“发”(类似门铃的喇叭发出什么声音)。

触发器的“触”称为动作特点,触发器的“发”便是逻辑功能。因此,触发器的逻辑功能与动作特点是两个不同的概念。触发器的逻辑功能由其特征方程描述,而触发器的动作特点(外部如何输入才能被触发器确认)则由触发器的电路结构决定。

从动作特点角度,如图 5.1.1(a)所示电路尽管具有触发器的基本功能,状态转移真值表也与 RS 触发器状态转移真值表吻合,但其状态变化不是在时钟信号作用下发生的,因存在着严重不足,难以满足实际要求。

主要有以下两点:

(1) 在输入信号存在期间,其电平直接控制着触发器的状态。而在实际应用中,很多场合都要求触发器的输入信号只能在控制信号的作用下,才能对触发器进行触发。当控制信号末到来时,输入信号对触发器的状态不起作用,即不能触发,触发器将保持原来的状态不

变。显然，基本 RS 触发器不能实现这种逻辑要求。

(2) 输入端需要加以约束条件($RS=0$，即 R、S 不允许同时为 1)限制，使用不方便。

从电路结构角度，触发器有同步结构、主从结构、维持阻塞结构等多种类型。同步结构触发器在基本结构触发器的基础上增加了同步时钟信号的控制电路，确保了只有在时钟信号作用下触发器的状态才会变化。主从结构触发器包括主、从两个同步结构的触发器。维持阻塞结构的触发器实现了上升沿触发的动作特点。

5.4.2 同步结构触发器的动作特点



如图 5.4.1 所示为同步结构的 RS 触发器时序图的参考实例，由图可总结同步触发器的动作特点如下：

图中，当 CP=0(钟控信号 CP 无效)时，尽管输入发生变化，但触发器保持状态 0 不变(图 5.4.1 的第 5、6 时间单元)；当 CP=1(钟控信号 CP 有效)时，触发器随着输入的变化而发生变化，正常工作。

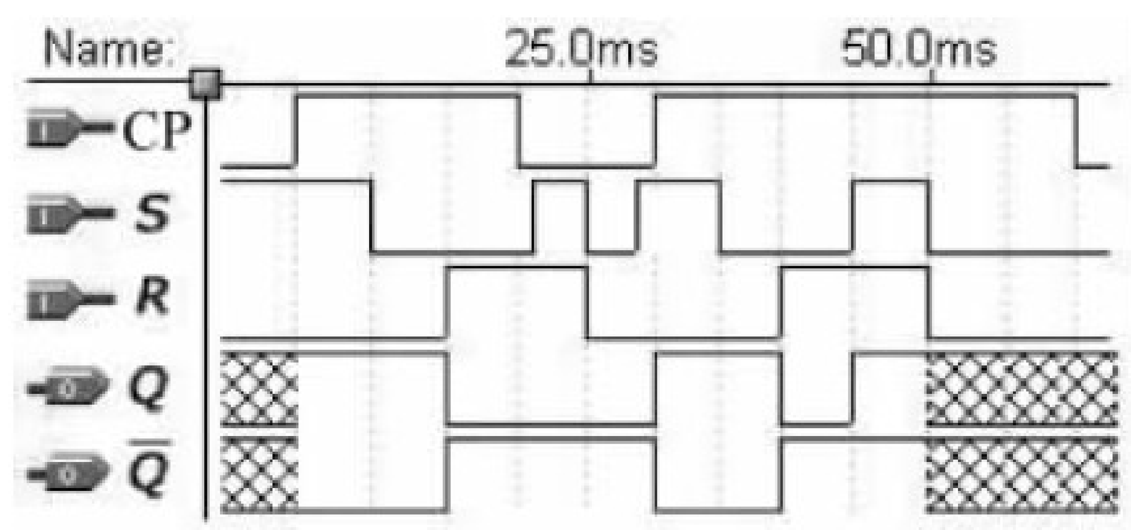


图 5.4.1　同步 RS 触发器时序图

如第 1 个 CP=1 区间，初始输入 $S=1$，$R=0$，触发器置 1，$Q^{n+1}=1(Q=1、\bar{Q}=0)$。之后，输入发生改变，$S=0$，$R=0$，触发器保持过去的状态不变，$Q^{n+1}=1$。之后，输入再次发生改变，$S=0$，$R=1$，触发器置 0，$Q^{n+1}=0$。

有兴趣的读者可进一步分析第 2 个 CP=1 区间，触发器状态随输入的变化情况。图中的第 4 个时间单元，$S=1$，$R=1$，输出 $Q=\bar{Q}=1$，既不是 1 状态，也不是 0 状态，之后，R、S 同时跳变到 0，输出状态不定。

同步触发器的动作特点：当钟控信号 CP 未到来时，同步触发器不接收输入信号，触发器的状态保持不变。当钟控信号 CP 到来时，触发器接受输入信号，正常工作，这种时钟控制方式称为电平触发方式。

当然，上面的动作特点是通过电路结构来保证的。如图 5.4.2 所示为同步 RS 触发器的电路图，它是在如图 5.1.2(a)所示基本 RS 触发器的基础上，增加了由与非门 G1、G2 组成的输入控制电路。右边部分为基本触发器，左边部分为钟控信号控制输入的控制电路。

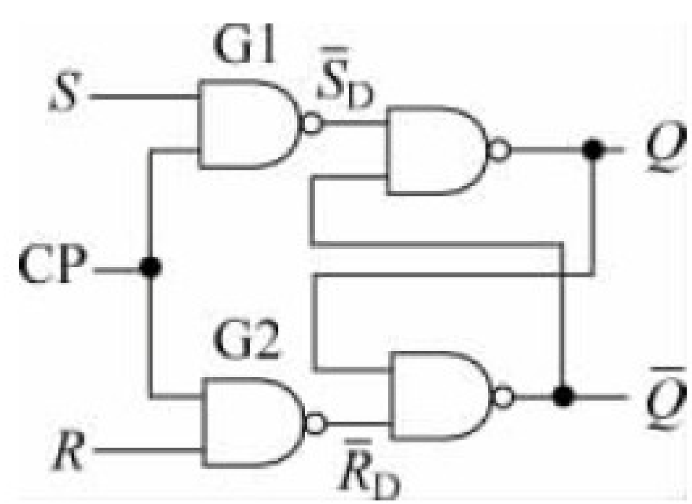


图 5.4.2　同步 RS 触发器电路

由如图 5.4.2 所示电路可分析出，当 CP 有效，$S=1$，$R=1$

时，输出 $Q=\bar{Q}=1$，应尽量避免。可见，同步 RS 触发器依然应遵循 $RS=0$ 的输入约束。

当然，同步 RS 触发器依然是 RS 触发器，具有如式(5.2.1)所示的特征方程描述的逻辑功能。有兴趣的读者，可自行通过该电路求出该电路的特征方程，即式(5.2.1)。

显然，同步触发器是指触发器的结构，即工作方式。就触发器的逻辑功能而言，同步触发器分为同步 RS 功能触发器、同步 JK 功能触发器、同步 D 功能触发器、同步 T 功能触发器等。

如图 5.4.3 所示为同步 D 功能触发器电路及逻辑符号。它是在如图 5.1.2(a)所示基本 RS 触发器的基础上，增加了由与非门 G1、G2 组成的控制电路。

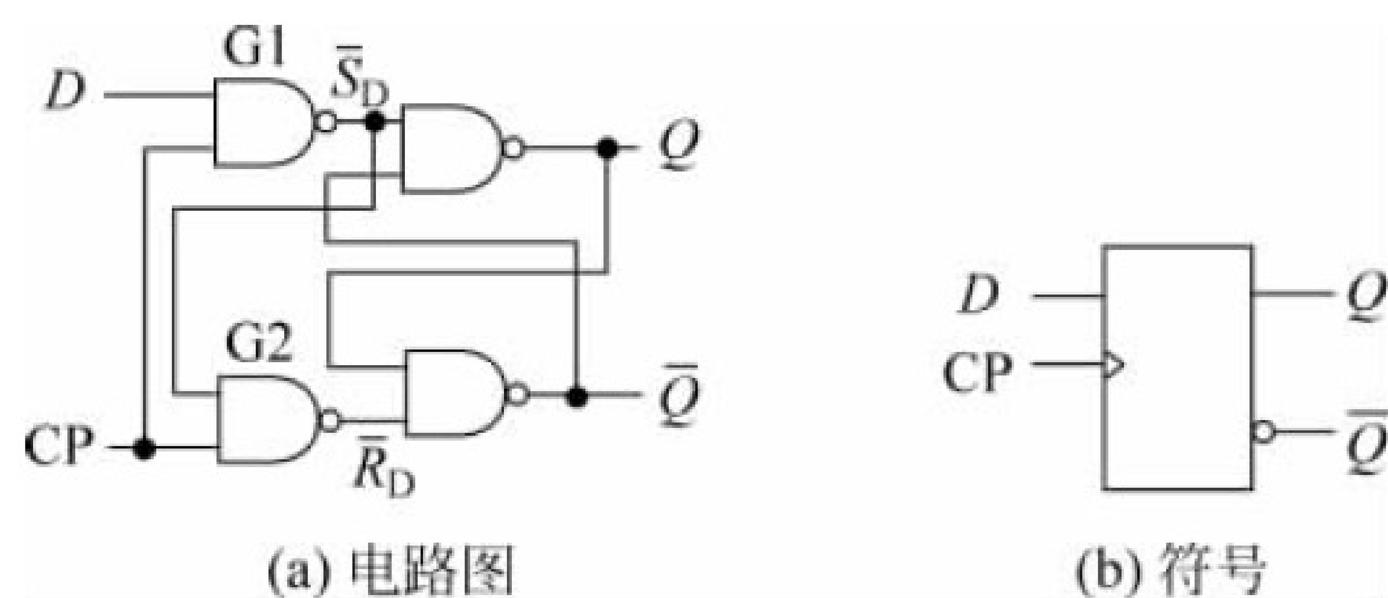


图 5.4.3 同步 D 触发器

如图 5.1.2(a)所示基本 RS 触发器的输入约束为

$$\bar{R}_D+\bar{S}_D=1$$

由电路可写出

$$\bar{R}_D=\overline{\bar{S}_D\cdot CP}=\overline{\overline{D\cdot CP}\cdot CP}=\overline{\bar{D}\cdot CP},\quad \bar{S}_D=\overline{D\cdot CP}$$

有

$$\bar{R}_D+\bar{S}_D=\overline{\bar{D}\cdot CP}+\overline{D\cdot CP}=\overline{\bar{D}\cdot D\cdot CP}=\overline{0\cdot CP}=1$$

可见，同步 D 功能触发器始终满足基本 RS 触发器的输入约束，故不需要对输入端加以约束。

当然，不管是同步 RS 触发器还是同步 D 触发器，它们均为电平触发方式。当钟控信号 CP 到来时，触发器接收输入信号，而且在此期间只要输入激励信号一旦有变化，都会引起触发器状态的改变。同步触发器在一次钟控信号 CP 有效期间，由于输入激励信号的变化引起触发器的状态发生两次或两次以上的转移，这种现象称为**触发器的空翻现象**。例如，图 5.4.1 所示波形中，第 2 个钟控信号 CP 有效期间，由于 R、S 的变化，使触发器出现空翻。

为了从根本上克服触发器的空翻现象，可采取其他电路结构，如主从结构、维持阻塞结构等。

5.4.3 主从结构触发器的动作特点

为了实现在钟控信号 CP 到来期间，触发器的状态只能改变一次，在同步触发器的基础上又设计了主从触发器。同理，主从触发器也是指触发器的结构，即工作方式。就触发器的逻辑功能而言，主从触发器也可分为主从 RS 功能触发器、主从 JK 功能触发器等。



主从结构的触发器包括主、从两个同步结构的触发器。如图 5.4.4 所示为主从结构的 RS 触发器时序图的参考实例。如图 5.4.5 所示为主从结构的 RS 触发器的参考电路，包括主、从两个同步结构的 RS 触发器。

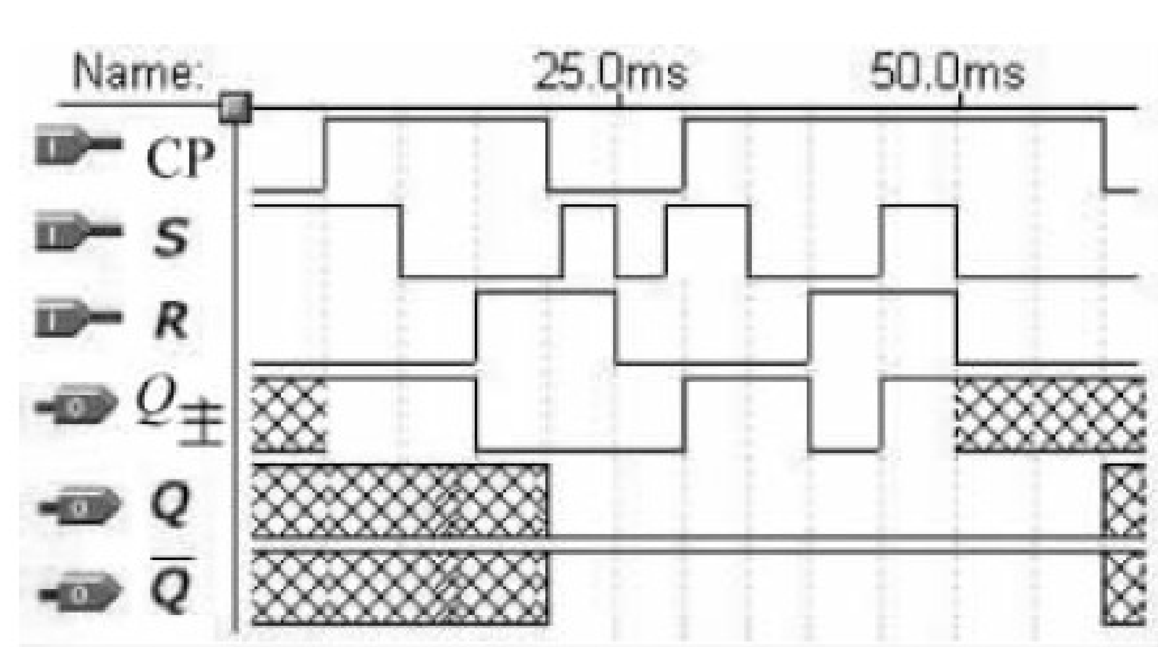


图 5.4.4　主从 RS 触发器时序图

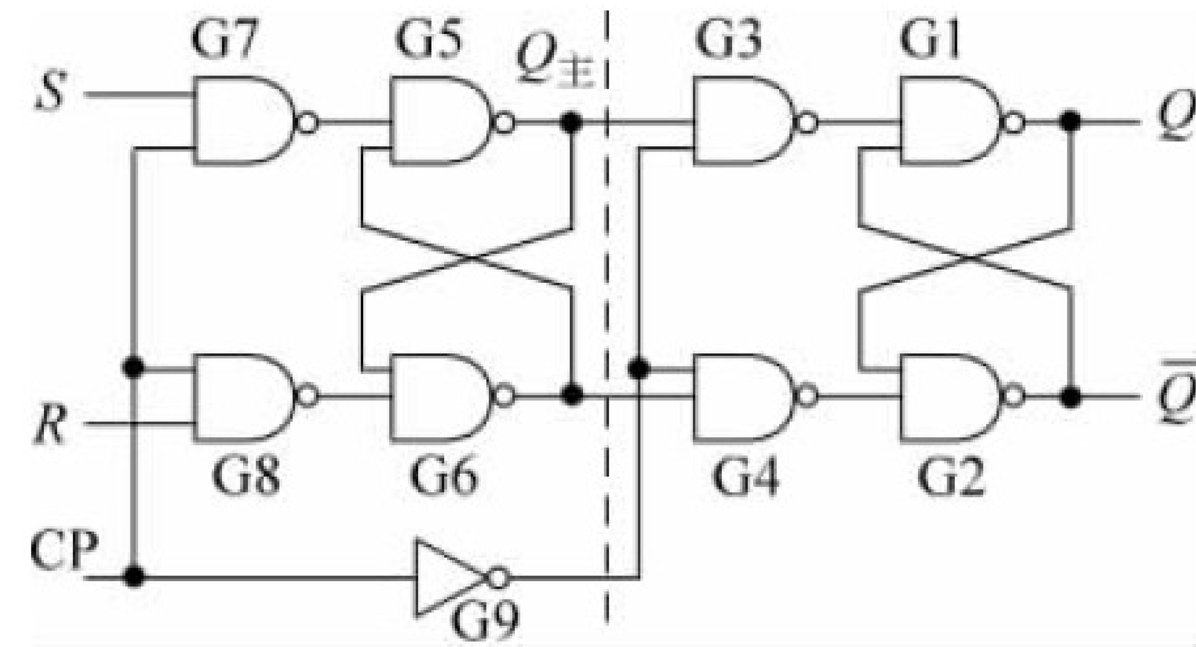


图 5.4.5　主从 RS 触发器电路

在如图 5.4.4 所示时序图中，第 1 个钟控信号 CP 高电平期间，主触发器接收输入信号，按照如图 5.4.1 所示的同步 RS 触发器发生变化，输出 $Q_主$具有和如图 5.4.1 所示的同步 RS 触发器的输出 Q 相同的波形。此时，由于主触发器的钟控信号 CP 取反后接从触发器钟控信号，作为最终输出的从触发器钟控信号为 0，保持初始状态不变。

在第 1 个钟控信号 CP 下降沿及低电平期间，主触发器的钟控信号 CP 为 0，保持钟控信号高电平期间最后一个状态不变。此时，从触发器钟控信号为 1，从触发器按照主触发器最后一个状态翻转到 0。

可类似分析第 2 个钟控信号周期主触发器及从触发器状态变化情况。

参考如图 5.4.4 所示主从 RS 触发器时序图，可总结主从结构触发器动作特点如下：

第 1 步，在 CP＝1 期间，主触发器接收输入信号，被置成相应状态，从触发器不工作。

第 2 步，在 CP＝0 期间，主触发器不工作，保持最后一个状态。从触发器按照主触发器状态翻转。

由于从触发器的输出为最终输出，因此，主从结构的触发器实现了在钟控信号 CP 到来期间，触发器的状态只改变一次。

必须指出，主从结构触发器的主触发器本身是一个同步触发器，存在着空翻问题。此外，主从结构的 JK 功能触发器的主触发器还具有一次变化现象。

如图 5.4.6 所示为主从 JK 触发器时序图参考实例，如图 5.4.7 所示为主从 JK 触发器电路。

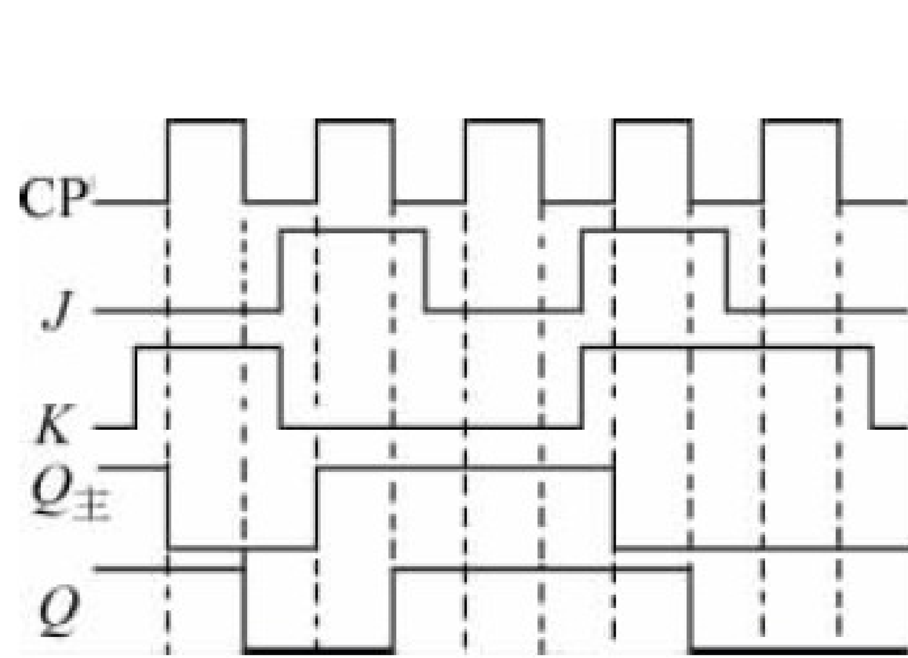


图 5.4.6　主从 JK 触发器时序图

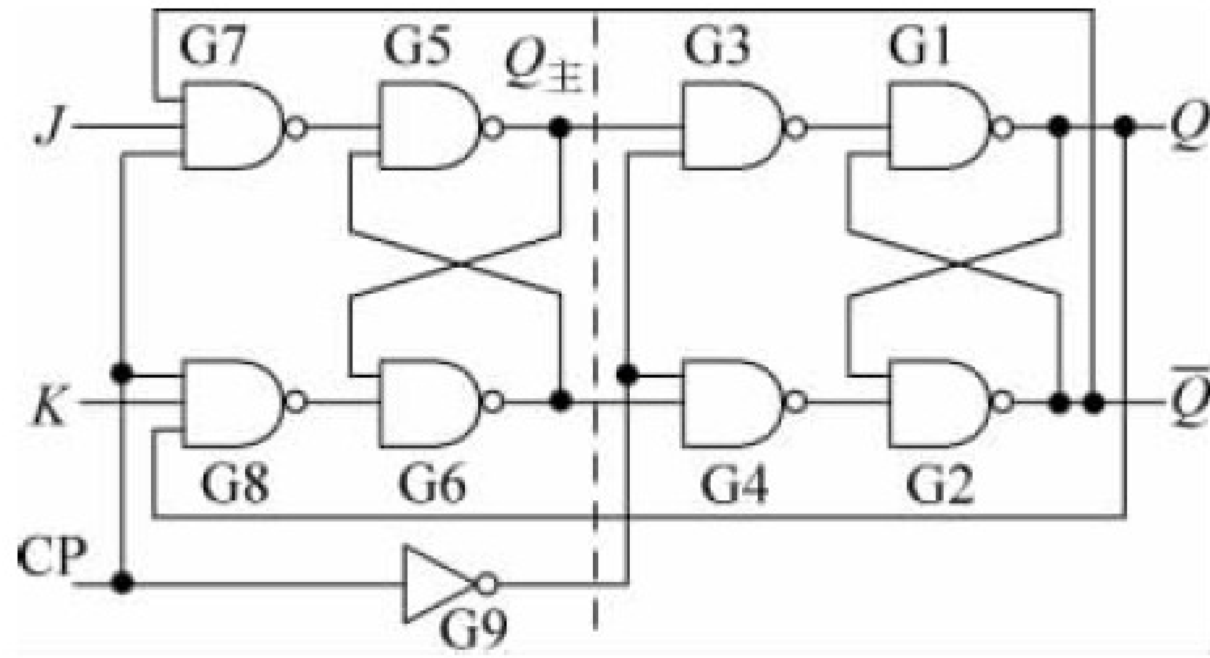


图 5.4.7　主从 JK 触发器电路

主从 JK 触发器为主从结构的触发器，具有主从结构触发器动作特点。在钟控信号 CP 高电平期间，主触发器接受输入信号，按照 JK 触发器逻辑功能发生变化。在钟控信号 CP 下降沿及低电平期间，从触发器跟随主触发器状态并相应发生变化。

如图 5.4.6 所示时序图中，在第 4 个钟控信号 CP 高电平期间，JK 触发器输入设置为状态翻转（$J=K=1$），主触发器状态立即由 1 翻转到 0。此时，输入并没有撤销，按照 JK 触发器逻辑功能，主触发器状态应该继续翻转，直到输入撤销或钟控信号无效。

由于主从 JK 触发器的主触发器状态只能一次变化，尽管主触发器由 1 翻转到 0 后钟控信号 CP 依然处于高电平，输入设置依旧为状态翻转，但主触发器不再由 0 翻转到 1。也就是说，在钟控信号 CP 上升沿及 CP=1 期间，主触发器的状态一旦发生变化，那么，无论输入信号如何变化，主触发器的状态也不会再发生变化，这种现象称为**主触发器的一次变化现象**。

当然，主从 JK 触发器主触发器的一次变化现象是由主从 JK 触发器的电路结构决定的。

对照主从 JK 触发器（图 5.4.7）、主从 RS 触发器（图 5.4.5）电路，不难发现，主从 RS 触发器的从触发器的输出对主触发器状态没有影响。主从 JK 触发器的从触发器的输出反馈到了输入端，将影响主触发器的状态改变。在钟控信号 CP 上升沿及 CP=1 期间，如果由于输入信号的变化使主触发器状态变化（即转移），则由于从触发器的状态并不发生变化，它的输出作为主触发器的输入将维持主触发器的状态不变。

例如，主从 JK 触发器初始状态为 0 状态（$Q=0$、$\overline{Q}=1$），在 CP=1 期间因为某种原因主触发器被置为 1 状态（$Q_主=1$、$\overline{Q}_主=0$）；那么，$Q=0$ 将使门 G8 输出为 1；而 $Q_主=1$ 和门 G8 的输出 1 将维持 $\overline{Q}_主=0$，从而维持 $Q_主=1$ 不变。可见，触发器初始 0 状态（$Q=0$）作为主触发器的输入正好维持主触发器的 1 状态不变，即无论输入信号如何变化，主触发器的状态也不会发生变化。

【例 5.4.1】 电路如图 5.4.8 所示，输入信号波形如图 5.4.9 所示，请画出对应的输出波形（两个 RS 触发器为具有异步功能的同步 RS 触发器）。

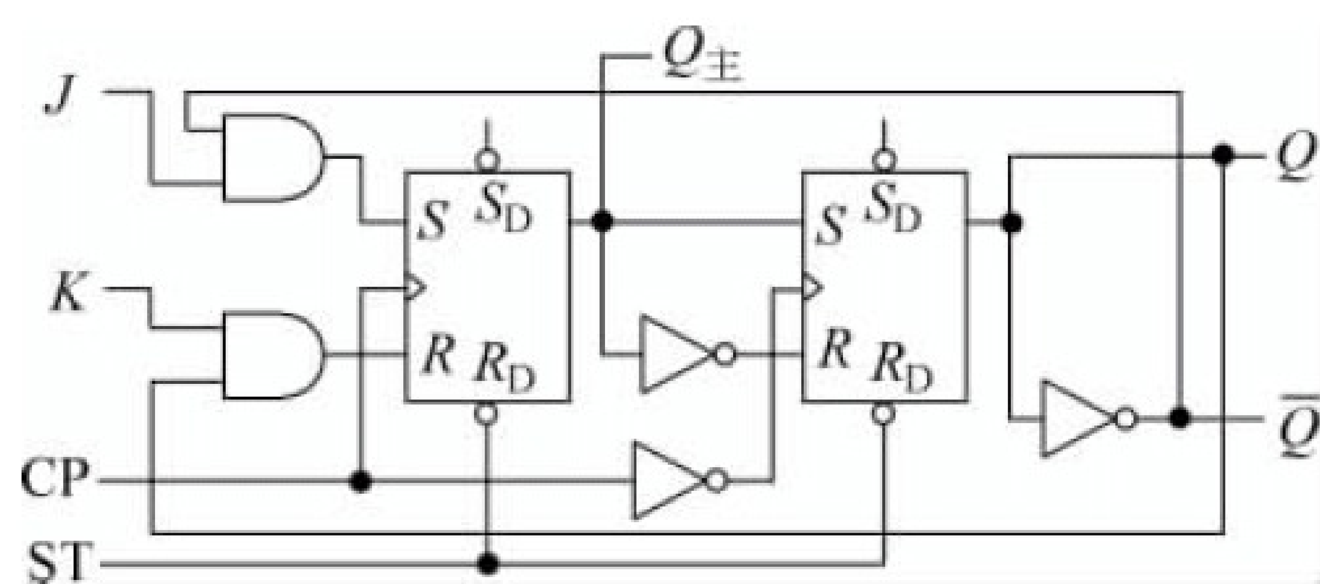


图 5.4.8 例 5.4.1 的图 1

解 （1）$\overline{S}_D$、$\overline{R}_D$ 为异步输入端。当 $\overline{R}_D=0$ 时，触发器立即复位到 0 状态；当 $\overline{S}_D=0$ 时，触发器被置位到 1 状态。其作用与时钟脉冲 CP 无关，故称为异步输入端。$\overline{R}_D$、$\overline{S}_D$ 端的小圆圈表示低电平有效。类似也存在具有异步功能的同步 D 触发器、JK 触发器等等。

（2）对照图 5.4.7，可知图 5.4.8 为具有异步功能的主从 JK 触发器。

（3）画出输出波形。第 1 个时间单元，ST＝0，主从 JK 触发器异步复位，主、从触发器状态均为 0。

第 1 个上升沿，主触发器工作，$J=1$，$K=0$，$Q_{主}=1$；

第 1 个下降沿，从触发器工作，按照主触发器状态翻转，$Q=1$。

类似地，可分析其他上升沿、下降沿时 $Q_{主}$、Q 的波形。输出波形如图 5.4.10 所示。

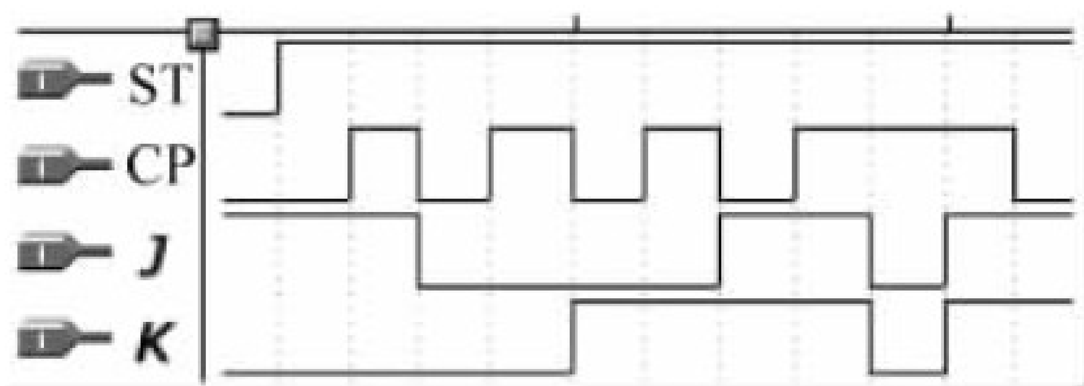


图 5.4.9 例 5.4.1 的图 2

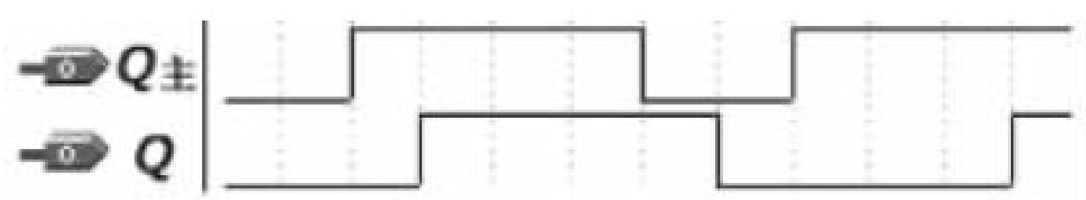


图 5.4.10 例 5.4.1 的图 3

主从 JK 触发器应用十分广泛，相应的集成芯片较多，单 JK 主从触发器有 74LS71、74LS72 等；双 JK 主从触发器有 74LS78 等。

5.4.4 边沿触发器的动作特点



为了克服主从 JK 触发器主触发器的一次变化问题，增强电路工作的可靠性，便出现了边沿触发器，如维持阻塞结构的触发器。

维持阻塞结构触发器的触发特点是只有在钟控信号 CP 的上升沿到来的瞬间，触发器才接收输入信号，称为上升沿触发。上升沿触发的 RS 触发器时序图实例如图 5.4.11 所示。图中，由于触发器只在 CP 信号上升沿到来瞬间接收输入信号，该时刻的输入均为 $S=1$，$R=0$，因此，触发器进入 1 并保持不变。

具有下降沿触发动作特点的 RS 触发器的时序图实例如图 5.4.12 所示。图中，由于触发器只在 CP 信号下降沿到来瞬间接收输入信号，因此，当第 1 个时钟下降沿到来时，输入 $S=0$，$R=1$，触发器进入 0 状态。当第 2 个时钟下降沿到来时，输入 $S=0$，$R=0$，触发器保持 0 状态不变。

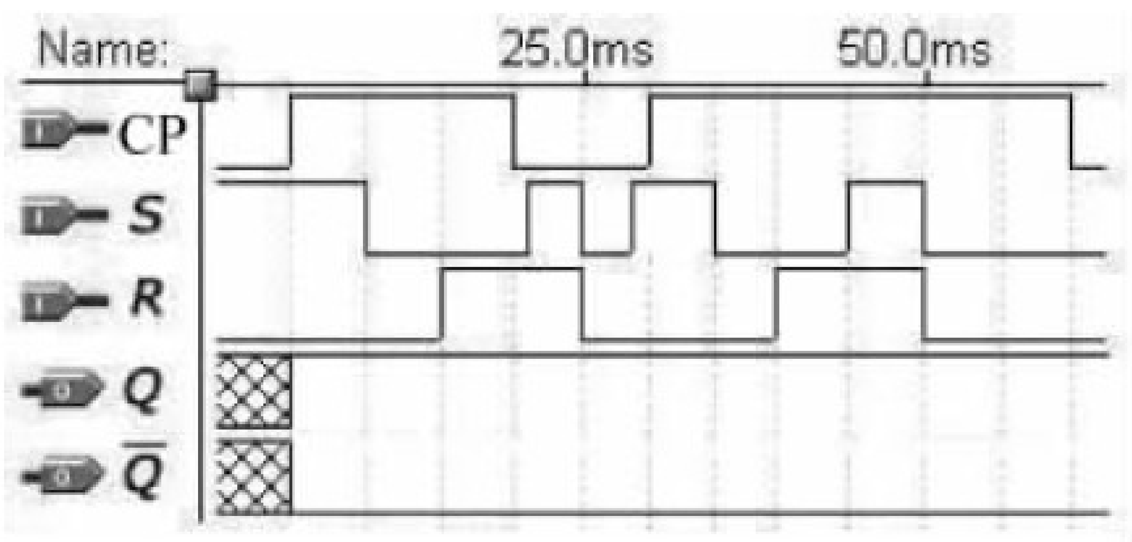


图 5.4.11 上升沿触发器时序图

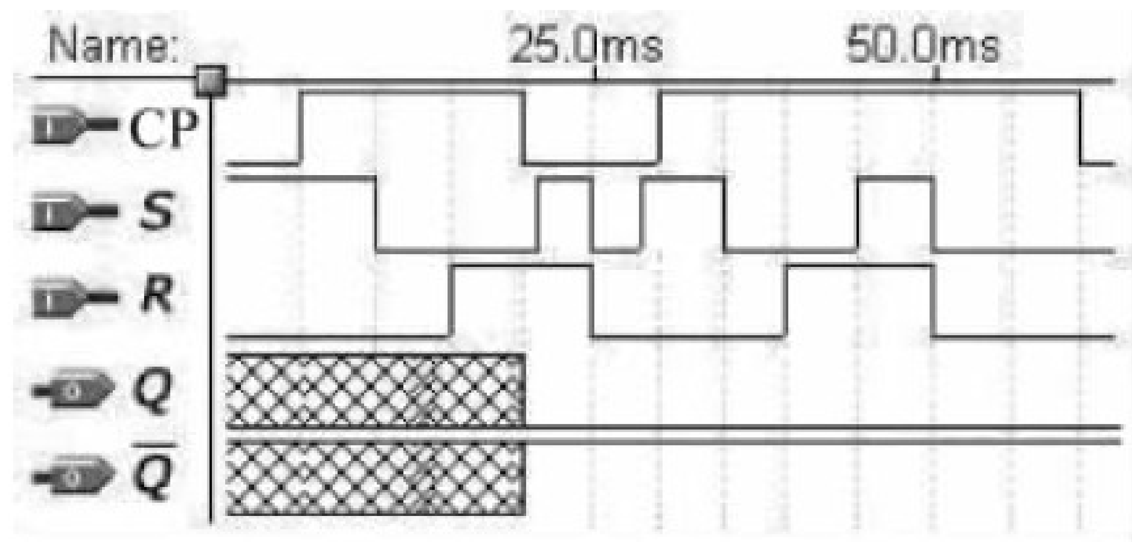


图 5.4.12 下降沿触发器时序图

综上所述，触发器的逻辑功能和动作特点是两个不同的概念，具有相同逻辑功能、不同动作特点的两个触发器在相同输入激励下其输出往往不相同。同步触发器动作特点为电平触发，只要控制电平有效，触发器即产生相应动作，最大的不足是可能产生空翻。主从结

构触发器动作分为两步：控制电平有效期间主触发器工作；控制电平无效时从触发器按照主触发器状态变化。主从JK触发器主触发器具有一次翻转性，因而降低了其抗干扰能力，尽管如此，其应用依旧十分广泛。边沿触发器在上升沿或下降沿到来时触发器动作，可靠性高，为目前时序逻辑电路的基本动作方式。需要说明的是，主从JK触发器从外部特性来看，与下降沿触发器非常类似(图5.4.6)，但触发器接收输入信号是在上升沿及CP=1期间，只是触发器最终状态翻转在下降沿。应注意它与下降沿触发器的区别。

有兴趣的读者可进一步了解边沿触发器动作特点的电路实现方法。

如图5.4.13所示为具有异步功能的维持阻塞D触发器的电路图，实现了上升沿触发的动作特点。同理，维持阻塞工作方式指触发器的结构，D触发器是指功能。与主从触发器相类似，也有维持阻塞RS触发器、维持阻塞JK触发器等。

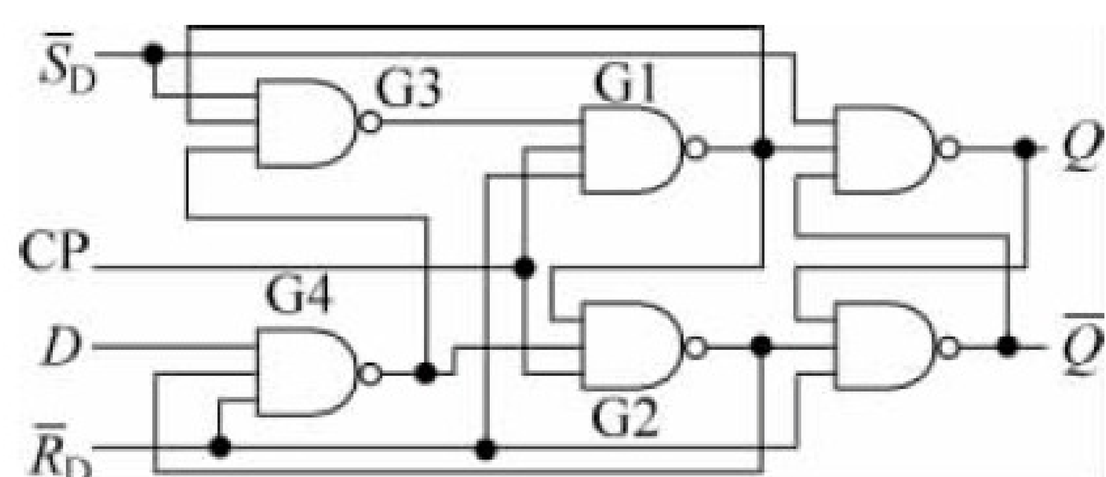


图5.4.13 维持阻塞D触发器电路

维持阻塞D触发器电路的工作原理如下：

1. $D=0$

当CP=0时，因为CP=0，门G1、G2的输出Y_1,Y_2为1，基本RS触发器维持原态。

当CP上升沿到来时，由于$D=0$，G4的输出$Y_4=1$；加之G1的输出$Y_1=1$反馈到G3，使门G3的输出$Y_3=0$；门G3的输出$Y_3=0$将维持门G1输出$Y_1=1$(G1的输出到G3的反馈线称为**维持线**)。同时，因为$Y_4=1$,$Y_1=1$,CP=1，使门G2的输出$Y_2=0$，触发器置0。又门G2的输出$Y_2=0$反馈到G4，使门G4截止，封锁输入D(G2的输出到G4的反馈线称为**阻塞线**)，使CP=1期间，无论信号如何变化，触发器状态不变。

2. $D=1$

当CP=0时，门G1、G2的输出Y_1、Y_2为1，使基本RS触发器维持原态。当CP上升沿到来时，由于$D=1$,$Y_2=1$，使门G4的输出$Y_4=0$；由于$Y_4=0$，门G2的输出$Y_2=1$；又由于$Y_4=0$，门G3的输出$Y_3=1$，加之CP=1使G1的输出$Y_1=0$，触发器置1。同理依靠置0维持线和置1阻塞线的作用可克服触发器空翻。

此外，如图5.4.13所示电路还具有异步功能。即当异步输入端工作时，CP、D均不起作用。如当$\bar{R}_D=0$,$\bar{S}_D=1$时，触发器立即复位，$Q^{n+1}=0$，即异步置0；当$\bar{R}_D=1$,$\bar{S}_D=0$时，触发器立即置1。显然，不允许出现$\bar{R}_D=\bar{S}_D=0$的情况。

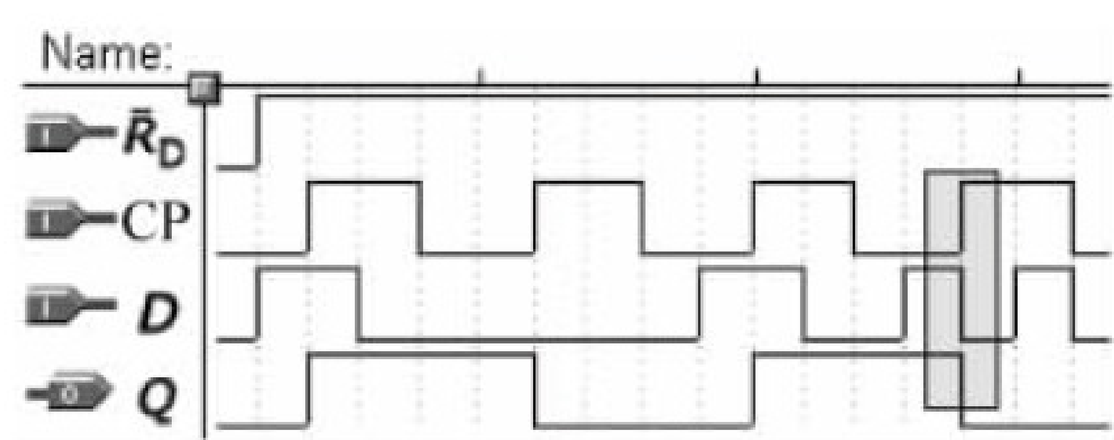


图5.4.14 维持阻塞D触发器的时序图

可见，如图5.4.13所示维持阻塞D触发器具有上升沿触发的动作特点，参考时序图如图5.4.14所示。由图易知维持阻塞D触发器的上升沿触发动作特点。

维持阻塞D触发器应用也十分广泛，相应的集

成芯片较多,如双维持阻塞 D 触发器 74LS74。

特别说明一点,在上升沿时刻,输入信号正好也发生了变化(图 5.4.14 阴影处),也产生了上升沿或下降沿,那么此时输入信号是变化前的状态还是变化后的状态?

图 5.4.14 是对图 5.4.13 的仿真,阴影处输入 D 的下降沿与 CP 上升沿同步到达,阻塞信号来不及阻塞输入的变化,触发器按变化后的输入信号状态发生变化。在实际应用中,应避免这种情况。

5.4.5 触发器的动作特点总结



通过本节的学习,读者应懂得触发器逻辑功能与动作特点是两个不同的概念,相同输入、相同逻辑功能的两个具有不同动作特点的触发器将可能得到差别很大的输出。

此外,通过时序图描述触发器,不仅可以反映触发器的逻辑功能,还能体现触发器的动作特点,因此,理解时序图是进一步应用时序电路的基础。如通过图 5.4.14 所示的维持阻塞 D 触发器的时序图可总结维持阻塞 D 触发器的基本使用方法如下:

第 1 步,设置输入。

第 2 步,给出上升沿。

常见结构触发器的动作特点汇总如表 5.4.1 所示。

表 5.4.1 常见结构触发器的动作特点

同步结构 RS 触发器	主从结构 RS 触发器
主要特点:CP 高电平期间接受输入,存在空翻现象	主要特点:CP 上升沿及高电平期间,主触发器接收输入信号,CP 下降沿及低电平期间,从触发器跟随主触发器状态发生变化。主触发器依旧存在空翻现象
主从结构 JK 触发器	**维持阻塞结构 RS 触发器**
主要特点:主触发器具有一次变化现象	主要特点:只在上升沿到来瞬间接收输入信号

*5.4.6 维持阻塞结构触发器的动态特性

为了保证触发器在工作时能可靠地翻转，了解触发器的动态翻转过程，找出翻转时对输入信号、时钟信号以及它们相互配合关系的要求是非常有必要的。

下面以维持阻塞结构 D 触发器为例介绍触发器的动态特性。

在如图 5.4.13 所示维持阻塞 D 触发器电路中，设所有门电路的平均传输延时相等，用 t_{pd}表示。

1. 输入信号宽度

图 5.4.13 中，$\bar{R}_D$、$\bar{S}_D$ 为异步输入端。当某个异步输入端有效时，如 $\bar{R}_D=1$，$\bar{S}_D=0$，触发器立即置 1，$Q=1$，$\bar{Q}=0$。上面的结论是忽略门电路的传输时间得出的。

如果考虑门电路的传输时间，那么不难发现，当 $\bar{R}_D=1$、$\bar{S}_D=0$ 时，触发器不会立即复位。忽略导线的传输时间，考虑门电路的传输时间，当 $\bar{S}_D=0$ 时，经过一个与非门的传输时间 t_{pd}，使 $Q=1$；再经过一个与非门的传输时间，使 $\bar{Q}=0$，触发器完成置 1 操作。

可求出异步输入端 $\bar{R}_D$、$\bar{S}_D$ 输入的低电平信号宽度为

$$t_w \geqslant 2t_{pd}$$

2. 建立时间

建立时间指输入信号先于 CP 动作沿到达的时间，用 t_{set}表示。

在如图 5.4.13 所示电路中，CP 信号直接加到 G1、G2 门的输入端。外部输入 D 输出经 G4 加到 G2 门的输入端，经 G4、G3 加到 G1 门的输入端。在 CP 信号上升沿到来前，G4、G3 的输出应该稳定，可求出建立时间为

$$t_{set} \geqslant 2t_{pd}$$

3. 保持时间

保持时间是指 CP 动作沿到达后，为确保触发器稳定动作，输入信号应保持的时间，用 t_H表示。

在如图 5.4.13 所示电路中，当 $D=0$ 时，CP 上升沿到来后，G2 门的输出依旧为 1，需要一个门电路传输时间才可变为 0，之后，反馈到 G4 门，封锁输入 D。因此，D 至少应保持一个门电路的传输时间，有

$$t_{HL} \geqslant t_{pd}$$

当 $D=1$ 时，CP 上升沿到来后，G1 门的输出为 0，锁定 G2 门的输出为 1，D 可立即撤销，有

$$t_{HH} \geqslant 0$$

4. 传输延时

顾名思义,传输延时是从输入信号到达起,到触发器输出稳定的新状态所经过的时间。对维持阻塞结构D触发器,其传输延时为从CP上升沿到来到触发器输出稳定的新状态所经过的时间。

由如图5.4.13所示电路不难看出,当初始状态为高电平,由高电平变为低电平时,只有$\overline{Q}=1$时,Q才可跳变到0,需要经历3个门电路的传输时间,因此有

$$t_{PHL}=3t_{pd}$$

可类似地分析出,当初始状态为低电平,由低电平变为高电平时,只需要经历一个门电路的传输时间,因此有

$$t_{PLH}=2t_{pd}$$

5. 最高时钟频率

维持阻塞结构D触发器为上升沿的触发器,因此,时钟信号的高电平持续时间应大于触发器的传输延时。因此,时钟信号高电平的宽度$t_{wH}\geqslant t_{PHL}=3t_{pd}$。

时钟信号低电平期间,应确保触发器状态保持不变(G1、G2输出为1,传输延时为t_{pd})。还应保证外部输入有足够的建立时间。因此,时钟信号低电平的宽度$t_{wL}\geqslant t_{pd}+t_{set}=3t_{pd}$。

可求出最高时钟频率为

$$f_{CPMAX}=\frac{1}{t_{wH}+t_{wL}}=\frac{1}{t_{PHL}+t_{pd}+t_{set}}=\frac{1}{6t_{pd}}$$

必须指出,对于实际的集成触发器,每个门的传输延时可能是不同的。此外,实际集成电路内部的逻辑门一般采用了各种形式的简化结构,其传输延时会比标准的输入、输出结构门电路小得多。上面介绍的动态参数及其求解过程只是动态分析方法的一种参考,实际产品的动态参数值,最终应通过实验测定求出。

复习与思考

5.4.1 什么是主从JK触发器的一次变化现象?可如何克服?

5.4.2 有人说主从JK触发器是下降沿的触发器。这种说法是否正确?为什么?

习题

5-1 填空题

1. 能够存储1位二值(逻辑0和逻辑1)信号的基本单元电路,统称为________。触发器具有两个输出端,当____________________时,触发器处于0状态。

2. 触发器在输入信号作用之前所处的原稳定状态称为________;触发器在输入信号

作用下所要进入的新的状态称为________。

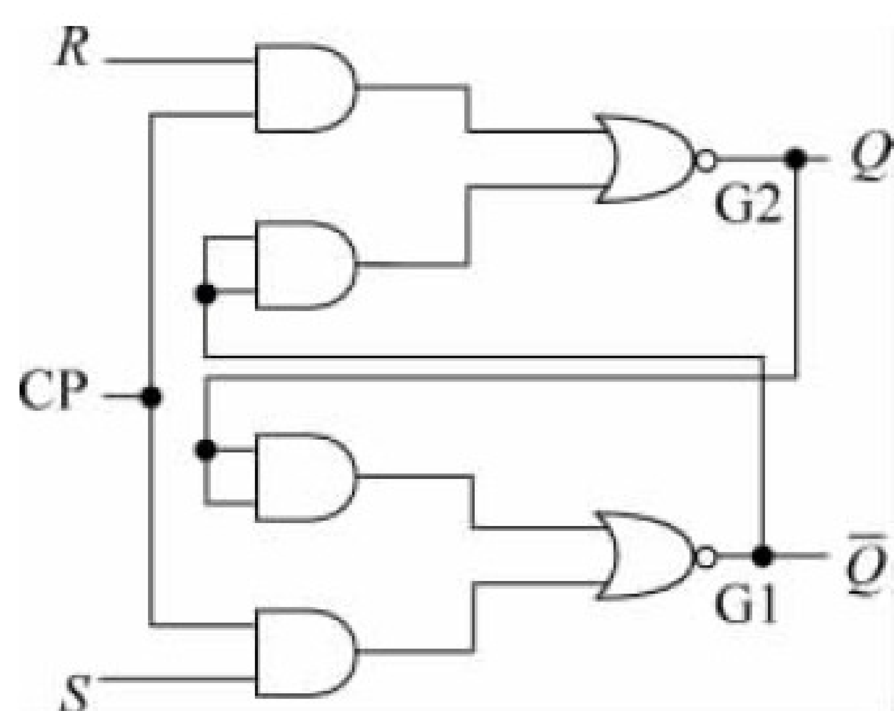
3. 对 RS 触发器，常用________描述 $R=S=1$ 的输入组合对应的输出状态。

4. 触发器的________与________是两个不同的概念，触发器的逻辑功能由其________描述，而触发器的________则由触发器的电路结构决定，而主从结构 JK 触发器的主触发器具有________________。

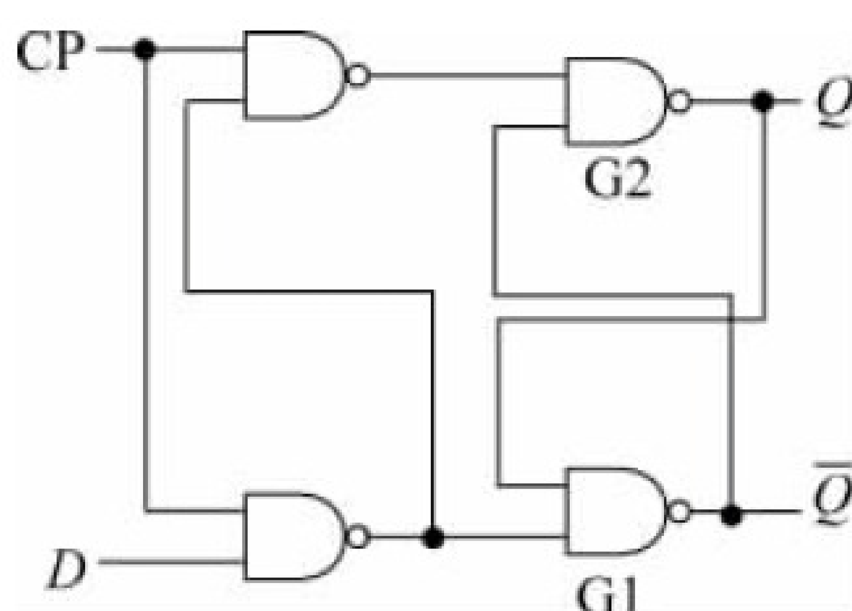
5-2　分析应用题

1. 分析如题图 5.1 所示触发器电路的逻辑功能，列出状态转移真值表，写出特征方程。

2. 分析如题图 5.2 所示触发器电路的逻辑功能，列出状态转移真值表，写出特征方程。



题图　5.1



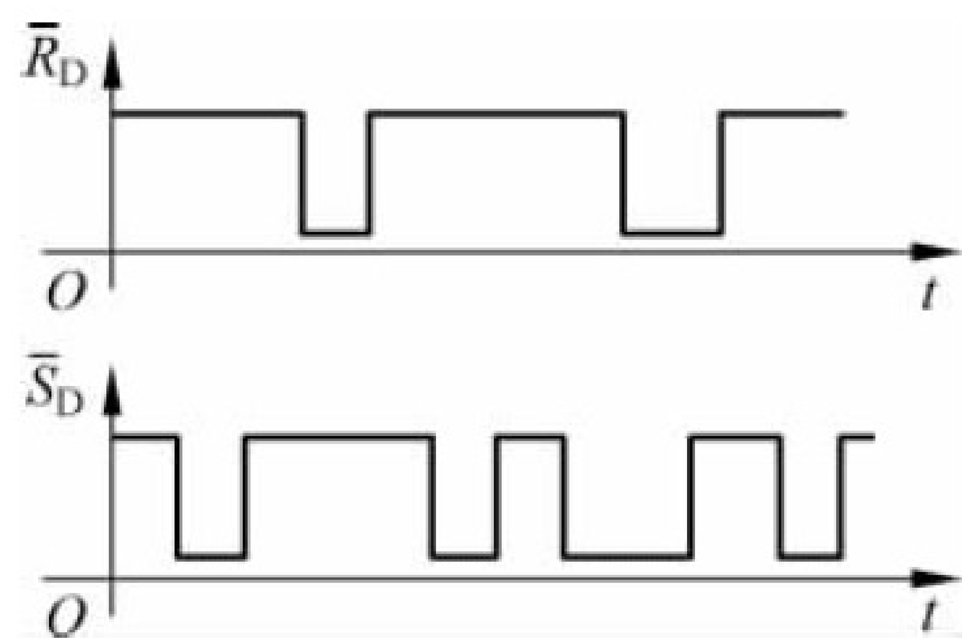
题图　5.2

3. 用 D 功能触发器实现 JK 触发器。

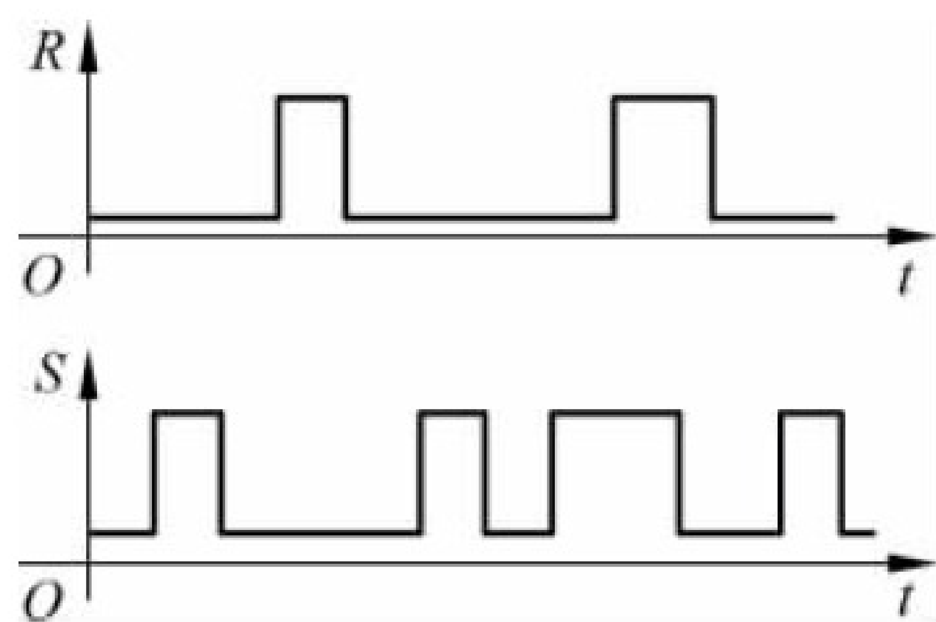
4. 已知一触发器的特征方程为 $Q^{n+1}=M\oplus N\oplus Q^n$，请用 JK 触发器实现该触发器的功能。

5. 将由与非门组成的基本 RS 触发器加上如题图 5.3 所示的输入 $\overline{R}_D$、$\overline{S}_D$ 波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

6. 将由或非门组成的基本 RS 触发器加上如题图 5.4 所示的输入 R、S 波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。



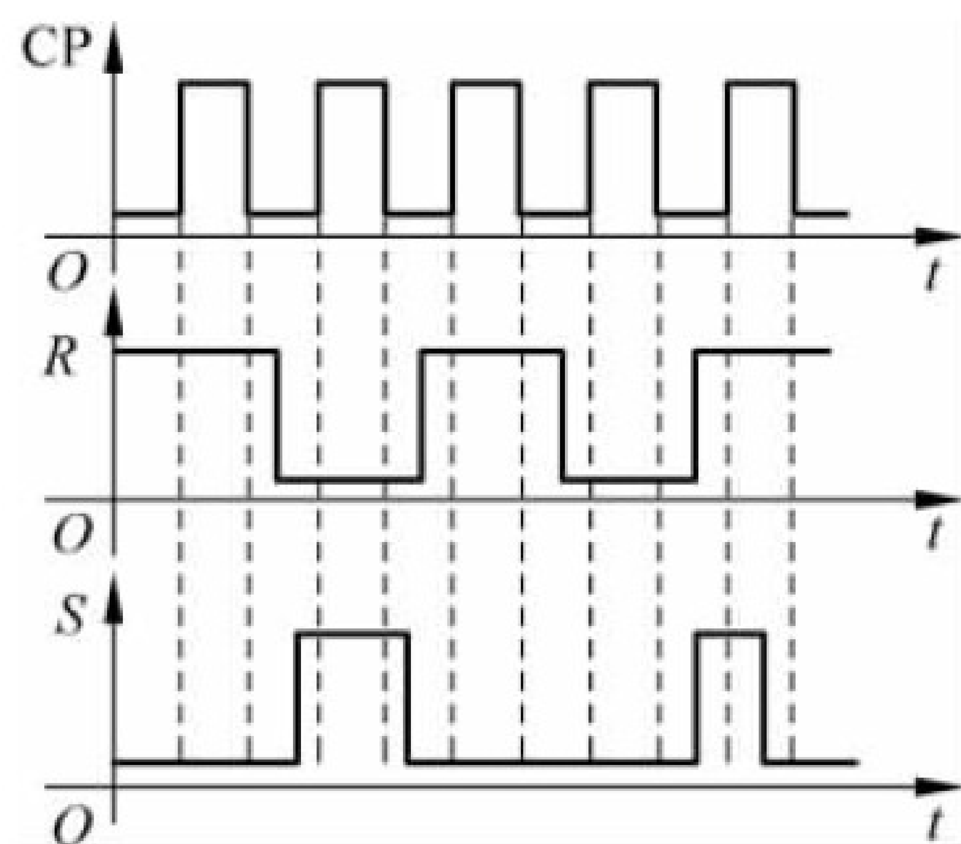
题图　5.3



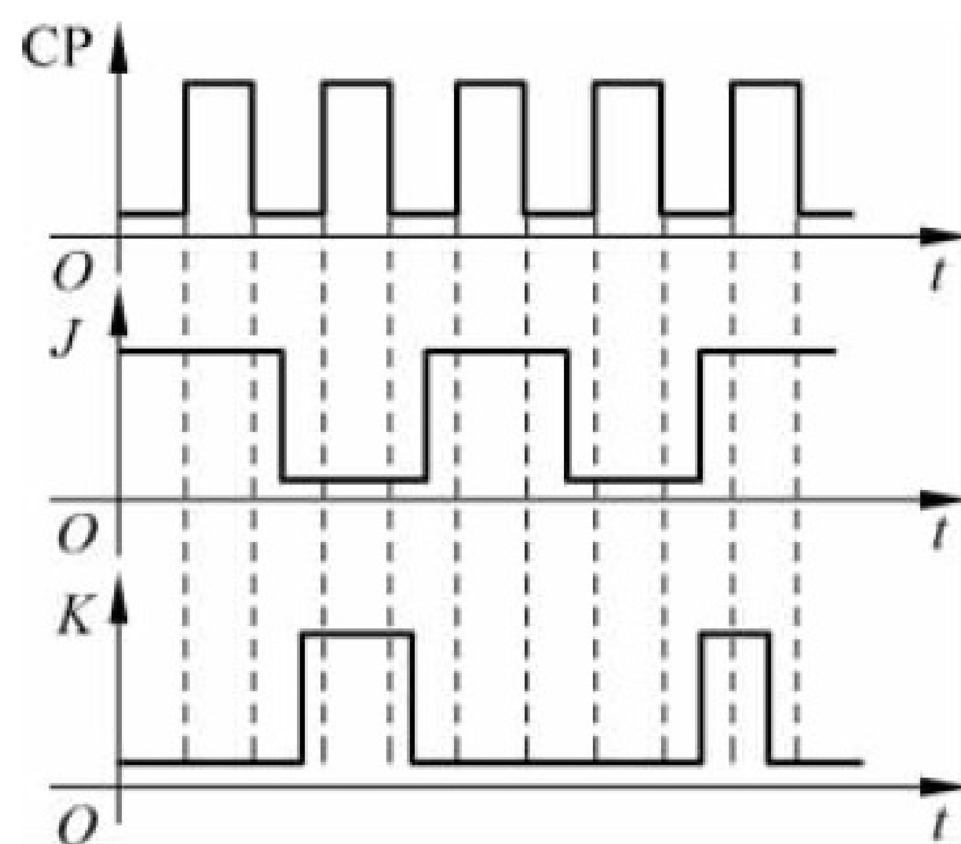
题图　5.4

7. 将同步结构 RS 触发器加上如题图 5.5 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

8. 将同步结构 JK 触发器加上如题图 5.6 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。



题图 5.5



题图 5.6

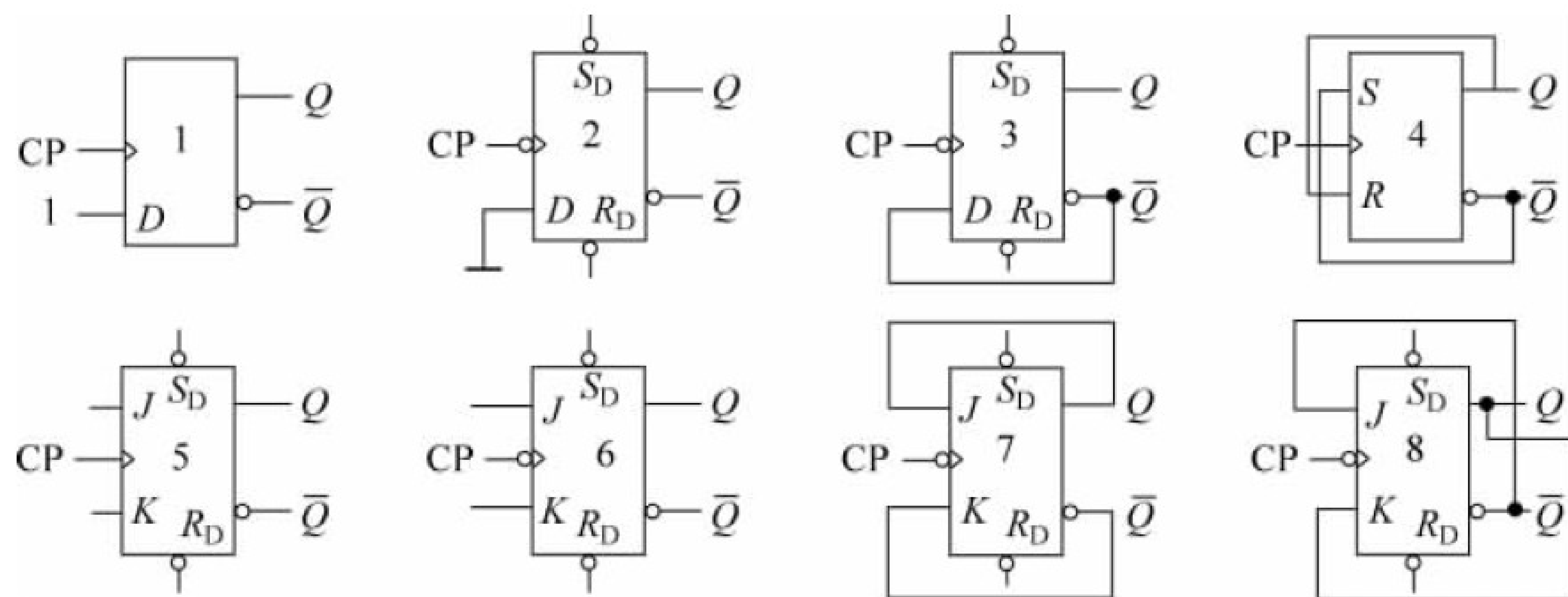
9. 将主从 RS 触发器加上如题图 5.5 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

10. 将主从 JK 触发器加上如题图 5.6 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

11. 将维持阻塞 RS 触发器加上如题图 5.5 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

12. 将维持阻塞 JK 触发器加上如题图 5.6 所示的输入波形，画出触发器 Q 端的波形(触发器初态为 0 状态)。

13. 已知如题图 5.7 所示电路中的各触发器的初始状态均为 0 状态，试对应画出在时钟信号 CP 的连续 5 个脉冲作用下各触发器输出端 Q 的波形。



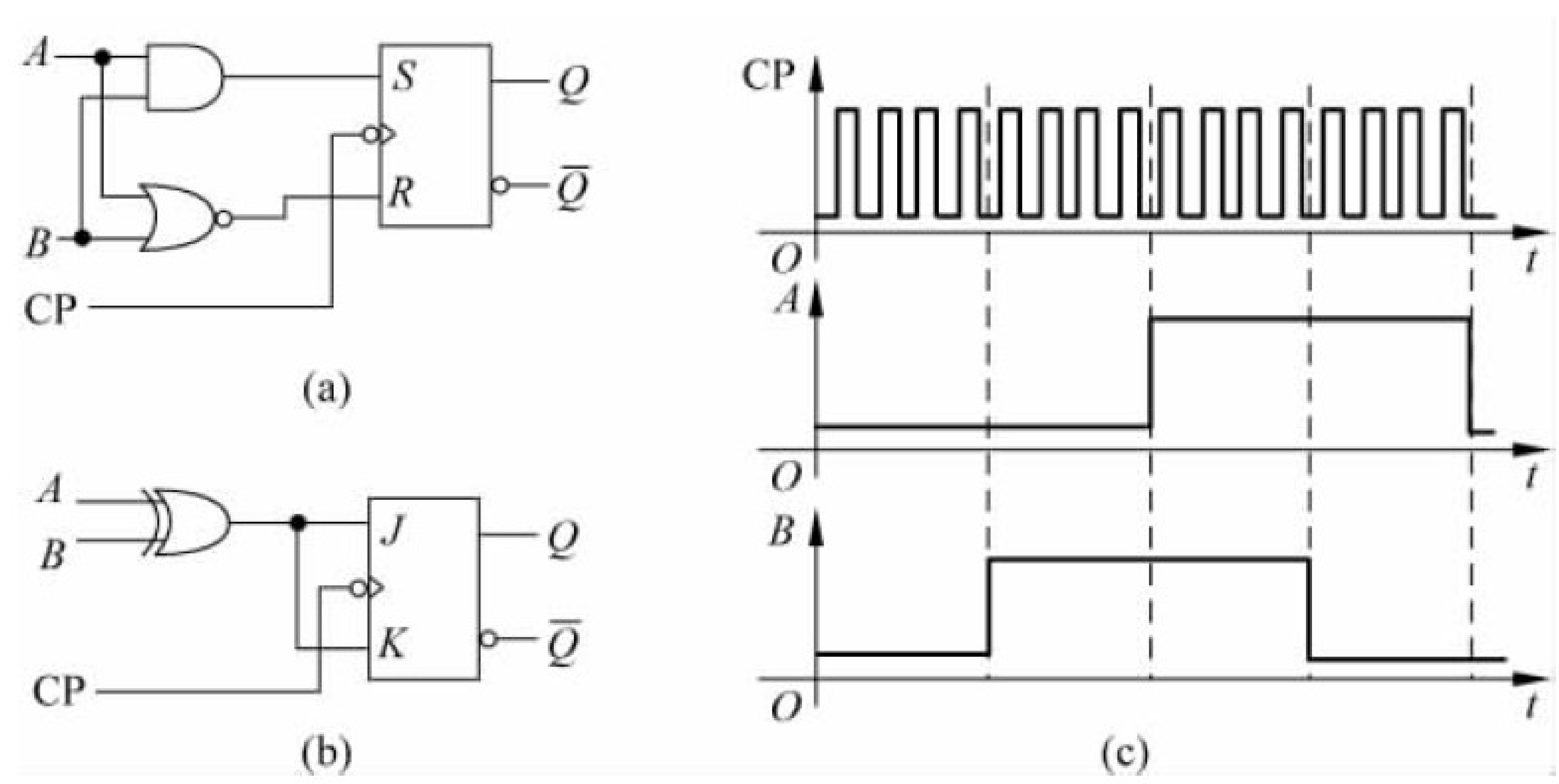
题图 5.7

14. 求如题图 5.8 所示各电路中的次态函数与输入、现态之间的逻辑关系式。

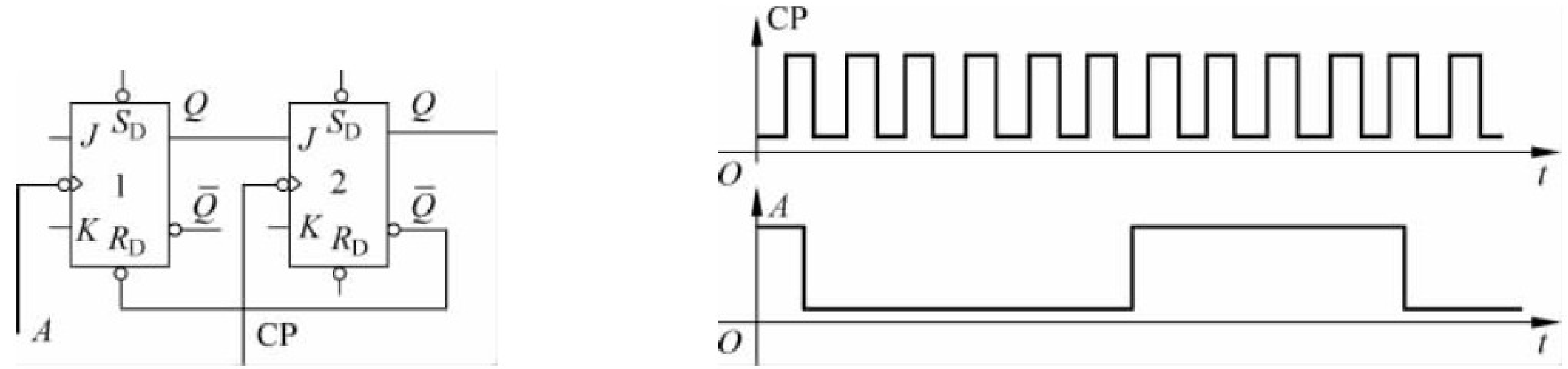
15. 如题图 5.8 所示各触发器的初始状态均为 0 状态，试对应画出在如题图 5.8(c)所示输入作用下各触发器输出端 Q 的波形。

16. 已知电路如题图 5.9 所示，主从 JK 触发器的初状态均为 0 状态，试根据如题图 5.10 所示波形画出触发器 2 的输出 Q 的波形。

17. 在分析应用题 16 中，若主从 JK 触发器 2 的 $\overline{Q}$ 不反馈到主从 JK 触发器 1 的复位端，输入不变，试画出触发器 2 的输出 Q 的波形。



题图 5.8

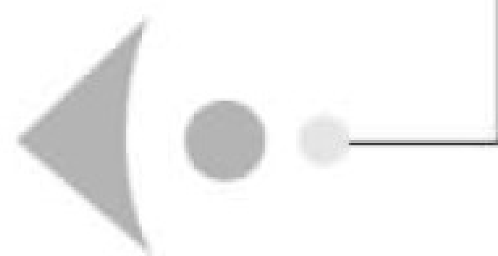


题图 5.9

题图 5.10



第6章 时序逻辑电路





本章要点：

本章为本书重点教学内容。学习本章应理解时序逻辑电路的概念、种类、描述方法及同步时序电路分析方法；理解寄存器、同步计数器等常用时序逻辑电路的概念、种类、电路特点及其常见集成电路的应用特点；初步理解异步时序电路的分析步骤、异步计数器的电路特点及其简单应用；了解同步时序电路设计的一般步骤，初步理解利用中规模集成电路解决实际逻辑问题的方法。

6.1 概述

第4章介绍了组合逻辑电路。本章介绍另一类数字电路——时序逻辑电路。

6.1.1 什么是时序逻辑电路



在组合逻辑电路中，任一时刻的输出信号仅取决于该时刻电路输入的信号，而与电路的原状态无关。对时序逻辑电路而言，任一时刻的输出信号不仅取决于该时刻电路的输入信号，而且还决定于电路原来的状态。具备这种逻辑功能特点的电路称为时序逻辑电路(Sequential Logic Circuit，时序电路)，以区别于组合逻辑电路。

根据上面的定义，交通信号灯、秒表等都属于时序逻辑电路。

下面简要解释为什么交通信号灯属于时序逻辑电路。假定信号灯当前倒计时时间为18s。读者可以思考一下，为什么倒计时时间为18s，而不是其他数据？因为刚才的倒计时时间为19s。可见，交通信号灯不仅与当前的输入信号有关，而且还与电路原来的状态有关，是典型的时序逻辑电路。

下面结合两个框图具体分析组合逻辑电路和时序逻辑电路的区别。

组合逻辑电路的框图如图6.1.1所示。显然，组合电路的输出(Z_i)仅与该时刻的输入

$(X_1, X_2, \cdots)$有关的主要原因是因为组合电路无存储单元,无记忆功能,没有记住之前的输出状态。

时序逻辑电路结构可以用图 6.1.2 所示的框图来表示。在电路结构上有两个显著的特点。第一,时序电路通常包含组合电路和存储电路两个组成部分,而存储电路是必不可少的。第二,存储电路的输出状态必须反馈到组合电路的输入端,与输入信号一起,共同决定组合逻辑电路的输出。可见,时序逻辑电路具有记忆功能,记住之前的输出状态,任一时刻的输出不仅取决于该时刻电路的输入,而且还与之前的输出状态有关,这是时序逻辑电路与组合逻辑电路的根本区别。

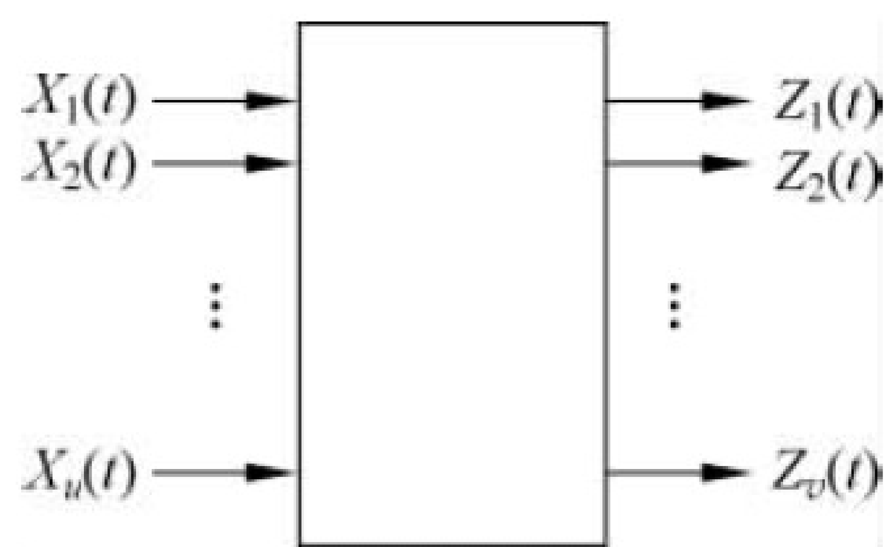


图 6.1.1 组合逻辑电路框图

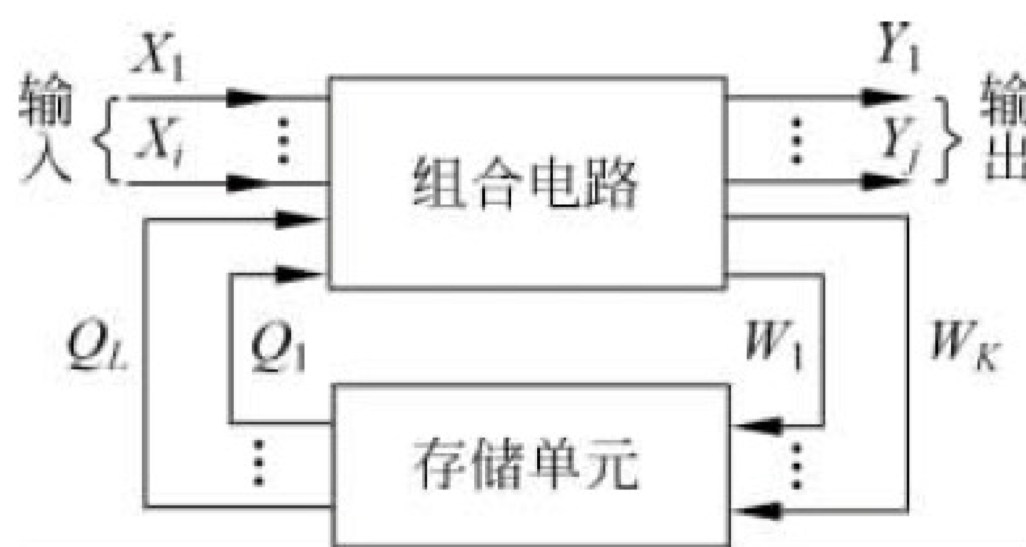


图 6.1.2 时序逻辑电路框图

6.1.2 时序逻辑电路的常见描述方法

可从如图 6.1.2 所示的时序逻辑电路的框图进一步理解时序逻辑电路的数学描述方法。

由如图 6.1.2 所示的时序电路示意框图可知,时序电路包括输入($X_1 \sim X_i$)、输出($Y_1 \sim Y_j$)、影响存储状态的存储电路输入($W_1 \sim W_k$)及存储电路的状态($Q_1 \sim Q_L$)等内、外部输入输出。因此,时序逻辑电路可以用输出($Y_1 \sim Y_j$)、存储电路的驱动($W_1 \sim W_k$)、存储电路的状态($Q_1 \sim Q_L$)等与输入、电路的现态关系的三组关系方程来描述。

对于如图 6.1.2 所示的时序电路,可以用 j 个输出方程、k 个驱动方程和 L 个状态方程来描述时序电路的逻辑功能。即

$$\begin{cases} Y_1 = F_1(X_1, \cdots, X_i, Q_1, \cdots, Q_L) \\ \vdots \\ Y_j = F_j(X_1, \cdots, X_i, Q_1, \cdots, Q_L) \end{cases} \tag{6.1.1}$$

$$\begin{cases} W_1 = G_1(X_1, \cdots, X_i, Q_1, \cdots, Q_L) \\ \vdots \\ W_k = G_k(X_1, \cdots, X_i, Q_1, \cdots, Q_L) \end{cases} \tag{6.1.2}$$

$$\begin{cases} Q_1^{n+1} = F_1(W_1, \cdots, W_i, Q_1^n, \cdots, Q_L^n) \\ \vdots \\ Q_L^{n+1} = F_j(W_1, \cdots, W_i, Q_1^n, \cdots, Q_L^n) \end{cases} \tag{6.1.3}$$

用方程来描述时序电路的逻辑功能,优点是根据方程画电路方便,缺点是不能直观地

看出电路的逻辑功能。

常用状态表、状态图、时序图描述时序电路，优点是可直观地看出时序电路的逻辑功能。

(1) 状态表。与触发器的状态表相同。只是这里已知的变量为电路输入 $X_1 \sim X_i$、电路的原状态 $Q_1 \sim Q_L$；待求为电路的新状态 $Q_1^{n+1} \sim Q_L^{n+1}$、存储电路的驱动 $W_1 \sim W_k$、电路的输出 $Y_1 \sim Y_j$。将它们用表格表示，即为状态转换真值表，简称状态表。

(2) 状态图。与触发器的状态图相同。即状态图中的小圆分别表示电路的各个状态，箭头表示状态转换的方向。同时，还在箭头旁注明电路状态转换前的输入变量取值和输出值。通常将输入变量取值在斜线以上，将输出值写在斜线以下。这种图形称为状态转换图，简称状态图。

状态图的优点是能直观、形象地表示出时序电路的逻辑功能。

(3) 时序图。所谓时序图，是根据状态表的内容，或者状态图的内容画成时间波形的形式。即在序列的时钟脉冲作用下，电路状态、输出状态随时间变化的波形图称为时序图。

用时序图描述时序电路的逻辑功能的优点是能够方便地用实验观察的方法来检查时序电路的逻辑功能。

6.1.3 时序逻辑电路的种类

时序电路的分类方法主要有如下两种。

(1) 按照时序电路中所有触发器状态的变化是否同步，时序电路可分为同步时序电路和异步时序电路。

通俗地讲，若电路中所有触发器的 CP 控制信号都使用同一个时钟脉冲，这种时序电路就称为同步时序电路；否则称为异步时序电路。

(2) 按照电路输出信号的特点，时序电路又可分为米利(Mealy)型时序电路和穆尔(Moore)型时序电路。

Mealy 型时序电路：其电路的输出信号不仅取决于存储电路的原状态，而且取决于电路的输入变量。其输出方程见式(6.1.1)。

Moore 型时序电路：其电路的输出信号仅仅取决于存储电路的原状态。其输出方程为

$$\begin{cases} Y_1 = F_1(Q_1, \cdots, Q_L) \\ \vdots \\ Y_j = F_j(Q_1, \cdots, Q_L) \end{cases} \tag{6.1.4}$$

实际上，Moore 型时序电路除时钟输入端外，没有其他的外部输入端。

应当指出，凡是符合时序电路含意的数字电路，都称为时序电路。常用的时序电路有寄存器、计数器、顺序脉冲发生器、检测器、读/写存储器等。

复习与思考

同学小王用手机重拨功能呼叫拨打小李手机，请问这个呼叫应用是组合电路应用还是

时序电路应用？为什么？

6.2 时序逻辑电路的分析

所谓时序电路的分析，就是根据已知的时序电路，找出该电路所实现的逻辑功能。具体地讲，就是要找出电路的输出及状态在输入变量和时钟信号作用下的变化规律。

6.2.1 同步时序电路的分析方法

描述时序电路逻辑功能的方法有方程式、状态表、状态图和时序图等。由于用状态表或者用状态图能直观地看出时序电路的逻辑功能，所以，在分析时序电路时，应设法找出该时序电路所对应的状态图或者状态表。

同步时序电路中所有触发器都是在同一个时钟脉冲作用下的，其分析方法相对简单，具体可按如下步骤进行分析。

(1) 根据给定的时序电路，写出电路的输出方程；写出每个触发器的驱动方程(又称为激励方程)。

(2) 将驱动方程代入相应触发器的特征方程，得到每个触发器的状态方程。

(3) 找出该时序电路相对应的状态表或者状态图，以便直观地看出该时序电路的逻辑功能。

(4) 若电路中存在着无效状态(即电路未使用的状态)应检查电路能否自启动。

(5) 文字叙述该时序电路的逻辑功能。

【例 6.2.1】 分析如图 6.2.1 所示电路的逻辑功能。

解 (1) 显然，这是一个由三个维持阻塞 JK 触发器组成的同步时序电路。它是一个 Moore 型的时序电路，因为该电路无输入信号作用。

(2) 写出电路的驱动方程、输出方程及状态方程。

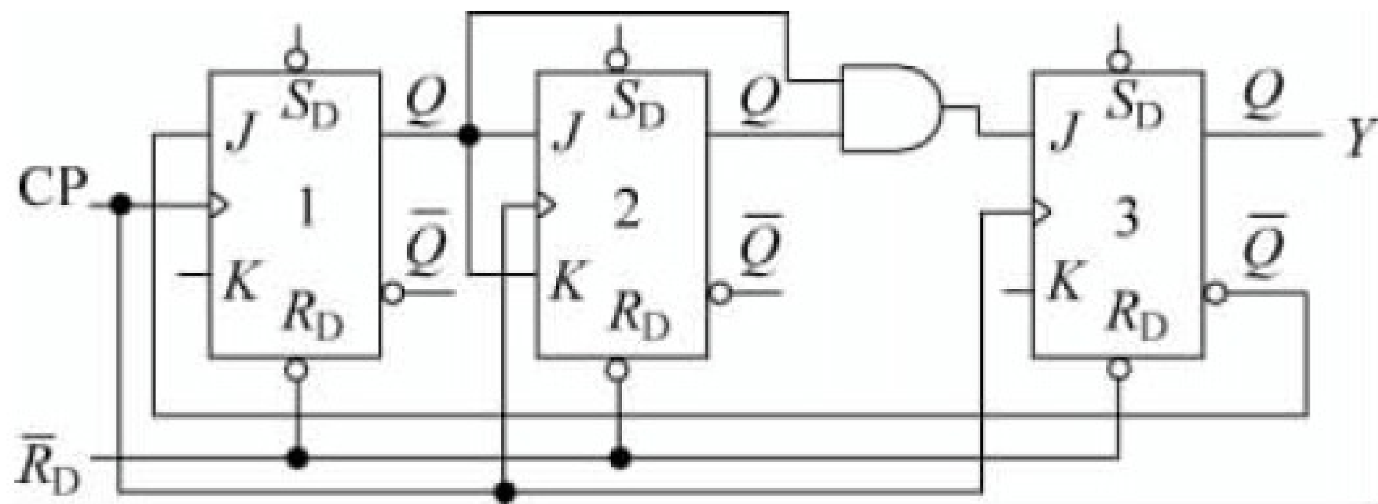


图 6.2.1 例 6.2.1 的图 1

驱动方程为(悬空为 1)

$$\begin{cases} J_1 = \bar{Q}_3^n, \quad K_1 = 1 \\ J_2 = Q_1^n, \quad K_2 = Q_1^n \\ J_3 = Q_1^n Q_2^n, \quad K_3 = 1 \end{cases} \tag{6.2.1}$$

输出方程为

$$Y = Q_3^n \tag{6.2.2}$$

将式(6.2.1)代入 JK 触发器的特性方程 $Q^{n+1}=J\bar{Q}^n+\bar{K}Q^n$ 中，得到电路的状态方程为

$$\begin{cases} Q_1^{n+1} = \bar{Q}_3^n\bar{Q}_1^n \\ Q_2^{n+1} = Q_1^n\bar{Q}_2^n + \bar{Q}_1^n Q_2^n = Q_1^n \oplus Q_2^n \\ Q_3^{n+1} = Q_1^n Q_2^n \bar{Q}_3^n \end{cases} \tag{6.2.3}$$

(3) 画出电路的状态图[①]。首先将电路清零。即在电路中各触发器的 $\bar{R}_D$ 端加一置 0 负脉冲，则该电路的状态“$Q_3^nQ_2^nQ_1^n$”为“000”。假设“000”为初始状态，当 CP 脉冲到来时，将电路的初始状态代入状态方程(即式(6.2.3))，可求出电路的新的状态为“001”，输出 $Y=0$。继续假定当前状态(现态)为“001”，代入状态方程，可求出电路的新的状态为“010”，输出 $Y=0$。

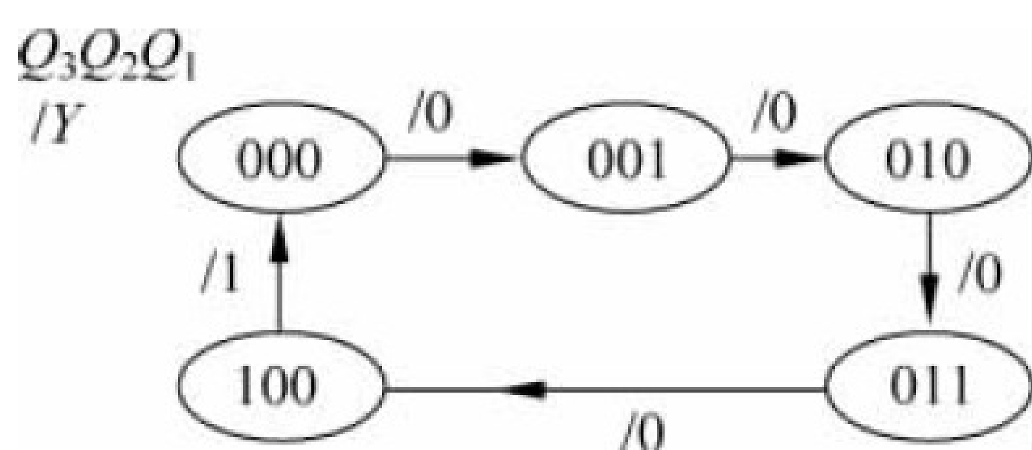


图 6.2.2　例 6.2.1 的图 2

以此类推，可得到如图 6.2.2 所示的状态图(同时还需求出输出 Y 的逻辑值)。

当然，读者可能会迷惑，当前状态(现态)为“011”时，电路的新的状态为“100”，输出 Y 为什么为 0 呢？Q_3 不是已经跳变到 1 了吗？

读者不要忘记，同步时序电路，所有触发器状态同时发生变化。电路当前状态为“011”，当时钟信号上升沿到来时，电路即将进入新的状态“100”，Q_3 并不是已经跳变到 1，而是即将跳变到 1，因此，图中标注输出 $Y=0$。

(4) 检查自启动。在图 6.2.1 中，电路用了 3 个触发器。电路应该有 $2^n=2^3=8$(n 为触发器数目)个状态。从状态图(图 6.2.2)中可以看出，电路只使用了 5 个状态 000、001、010、011、100，这 5 个状态称为有效状态。电路在 CP 控制脉冲作用下在有效状态之间的循环称为有效循环。该电路还有 3 个状态 101、110、111 没有使用，这 3 个状态称为无效状态。电路在 CP 脉冲作用下，在无效状态之间的循环，称为无效循环。

所谓电路能够自启动，就是当电源接通或者由于干扰信号的影响，电路进入无效状态，在 CP 控制脉冲作用下，电路能够进入有效循环，则称电路能够自启动；否则，电路不能自启动。



下面检查例 6.2.1 中的电路能否自启动。

设电路的初始状态为“101”，当 CP 控制脉冲到来时，将初始状态代入状态方程、输出方程，可求出输出为“1”，新状态为“010”；类似可得出电路的初始状态为“110”时，在 CP 脉冲作用下输出为“1”，新状态为“010”；电路的初始状态为“111”，在 CP 脉冲作用下输出为“1”，新状态为“000”。所以，电路能够自启动。故可以画出如图 6.2.3 所示的完整的状态图。

① 本例详细给出了通过特征方程求状态图的步骤。在后面的内容中，将直接给出状态图，其求出过程同本例。

显然，在图 6.2.3 中，电路由无效状态转换到有效状态过程中的输出 $Y=1$ 为无效输出。

（5）结论。从图 6.2.3 中很容易看出，每经过 5 个时钟信号，电路的状态循环变化一次，所以这个电路具有对时钟信号计数的功能。同时，因为每经过 5 个时钟脉冲作用以及输出端 Y 输出一个进位脉冲。因此，图 6.2.1 所示电路是一个能够自启动的同步五进制加法计数器。

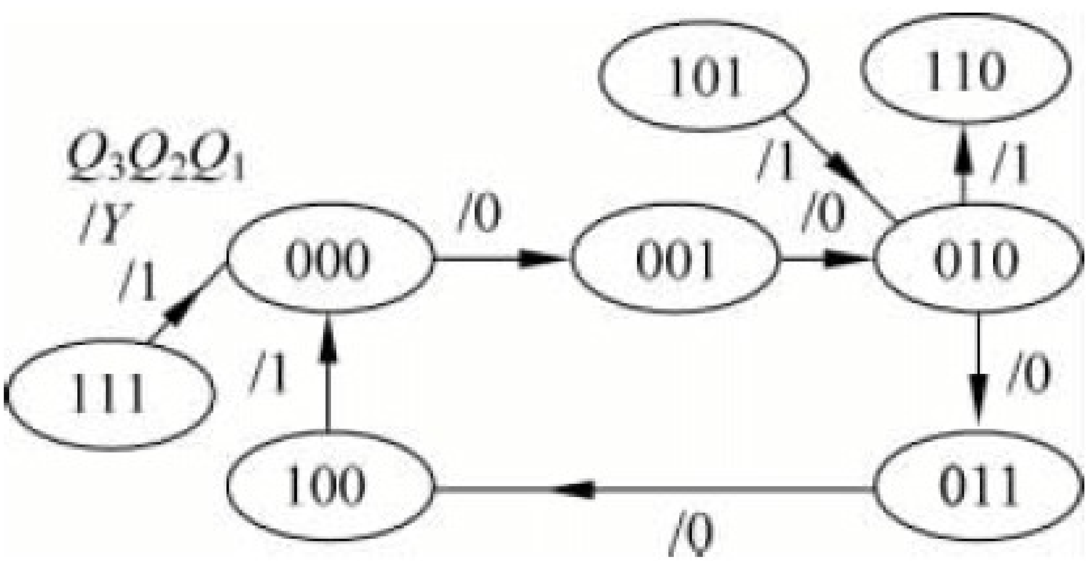


图 6.2.3　例 6.2.1 的图 3

表 6.2.1　状态转移真值表

CP	Q_3　Q_2　Q_1	$Q_3^{n+1}Q_2^{n+1}Q_1^{n+1}Y$
1↑	0　0　0	0　0　1　0
2↑	0　0　1	0　1　0　0
3↑	0　1　0	0　1　1　0
4↑	0　1　1	1　0　0　0
5↑	1　0　0	0　0　0　1
6↑	1　0　1	0　1　0　1
7↑	1　1　0	0　1　0　1
8↑	1　1　1	0　0　0　1

当然，也可以写出该电路相应的状态表。它可以由状态转换图转换得到；也可以依次假设电路的初始状态代入状态方程，由输出方程得到。状态表如表 6.2.1 所示。显然，用状态表描述该电路的逻辑功能不如状态图直观，在后面的分析中，若给出状态图便不再给出状态表。

（6）计算机仿真。如图 6.2.1 所示电路在 Quartus Ⅱ 环境中的仿真结果如图 6.2.4 所示。图中的数字输出形式为三个触发器按照 $Q_3Q_2Q_1$ 顺序（仿真图编号顺序为 $Q_2Q_1Q_0$）以总线形式的仿真结果。图中，$Q_3Q_2Q_1$ 从状态 0 开始，经 5 个时钟，历 4 个状态后回到 0，为对时钟信号计数的同步五进制加法计数器。所有状态均在时钟上升沿发生变化，为同步五进制上升沿计数的加法计数器。



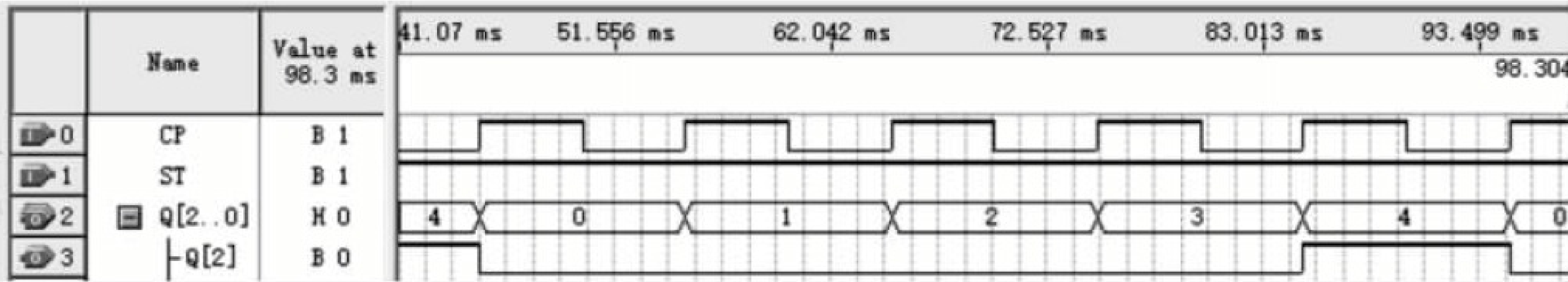


图 6.2.4　例 6.2.1 的图 4

对五进制自然规律加法计数器，最大值为 4（满幅值）。图 6.2.4 中，当进入状态 4 时，输出 Y（仿真图为 Q_2）跳变到 1，可见，图 6.2.1 所示电路为同步五进制上升沿计数满幅输出

上升沿的加法计数器。此外,当由状态 4 回到状态 0 时,计数器发生溢出,输出 Y 由 1 跳变到 0,输出下降沿。

综上所述,图 6.2.1 所示电路的逻辑功能及其主要特点如下:该电路为同步五进制上升沿计数满幅输出上升沿溢出输出下降沿的加法计数器。

可见,通过时序图分析时序电路,不仅可直观看出电路的逻辑功能,还能直观反映电路的应用特点。因此,读懂并理解电路的时序图是更好运用时序电路的基础。

【例 6.2.2】 分析如图 6.2.5 所示电路的逻辑功能。

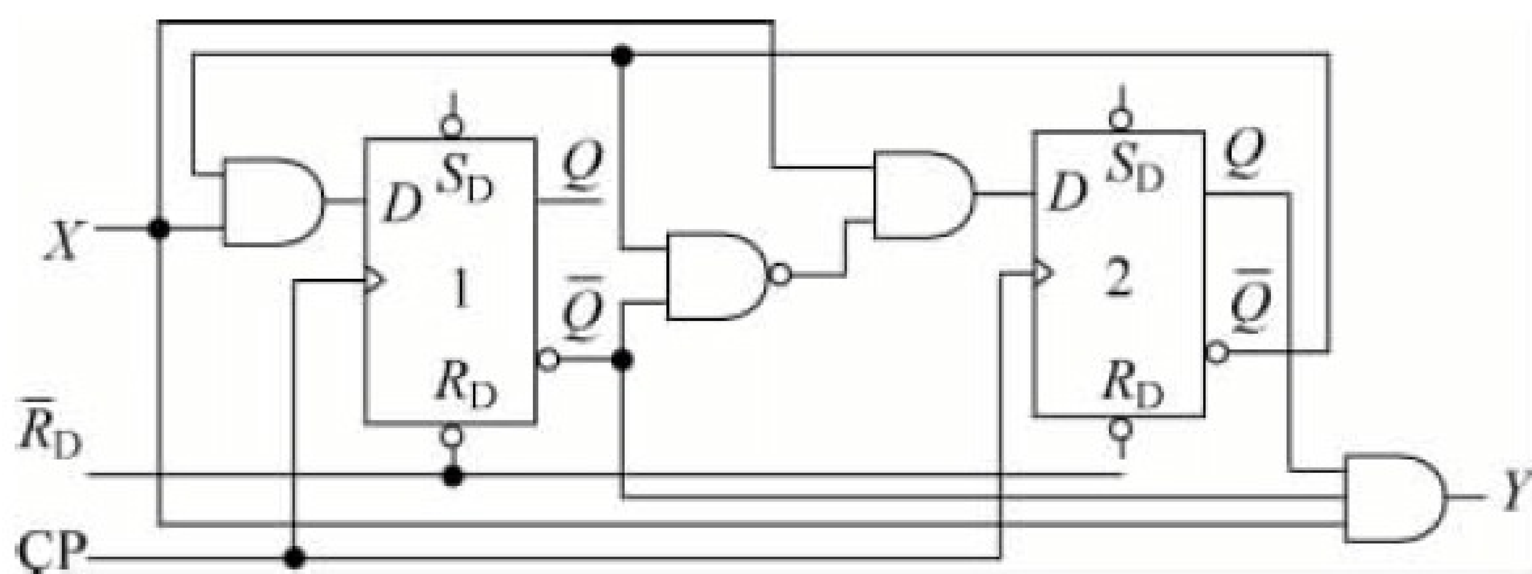


图 6.2.5 例 6.2.2 的图 1

解 (1) 显然,这是一个 Mealy 型的时序电路。它是由两个维持阻塞 D 触发器组成的同步时序电路。X 为电路的输入端,Y 为输出端。

(2) 写出电路的驱动方程、输出方程及状态方程。

驱动方程为

$$\begin{cases} D_1 = X\bar{Q}_2^n \\ D_2 = \overline{\bar{Q}_1^n \bar{Q}_2^n} X \end{cases} \tag{6.2.4}$$

输出方程为

$$Y = \bar{Q}_1^n Q_2^n X \tag{6.2.5}$$

将式(6.2.4)代入 D 触发器的特性方程 $Q^{n+1}=D$,得到电路的状态方程为

$$\begin{cases} Q_1^{n+1} = X\bar{Q}_2^n \\ Q_2^{n+1} = \overline{\bar{Q}_1^n \bar{Q}_2^n} X \end{cases} \tag{6.2.6}$$

(3) 画出电路的状态图。由状态方程可得到如图 6.2.6 所示的状态图。

求解方法:先假定电路的初始状态 Q_2Q_1 为"00",输入 $X=1$;将初始状态代入状态方程、输出方程,可求出当 CP 控制脉冲到来时,输出为"0",新状态为"01";继续求 $X=1$ 时,剩余各状态的状态变化规律。

参照上面的方法,进一步求出 $X=0$ 时的状态变化规律,可求出如图 6.2.6 所示的状态图。

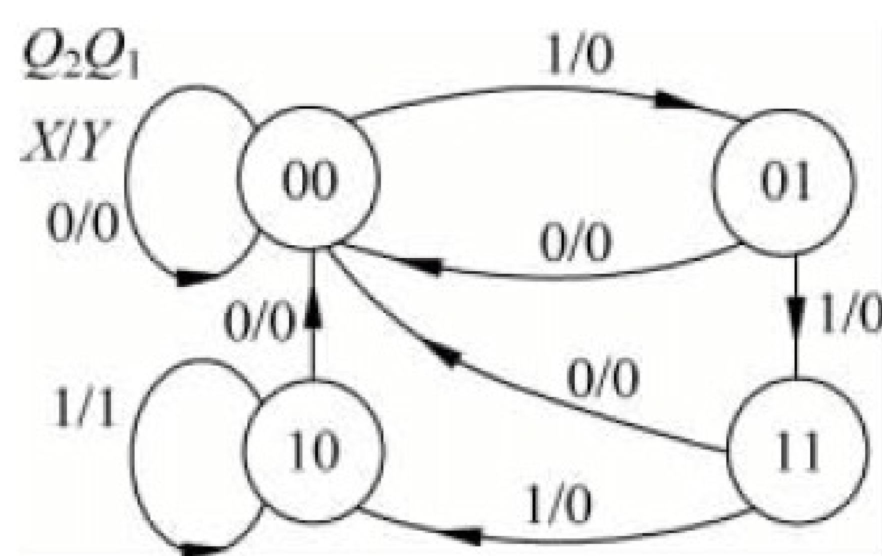


图 6.2.6 例 6.2.2 的图 2

(4) 结论。当输入端 X 连续输入 4 个或 4 个以上的 1 时,输出为 1,否则,输出为 0。可见,图 6.2.5 所示电路为同步的"1111"序列检测器。

(5) 计算机仿真[①]。如图 6.2.5 所示电路在 Quartus Ⅱ环境中的仿真结果如图 6.2.7 所示。从仿真图可看出如图 6.2.5 所示电路为同步的上升沿触发的“1111”序列检测器(仿真图给出了连续输入 3 个 1 的状态及输出)。

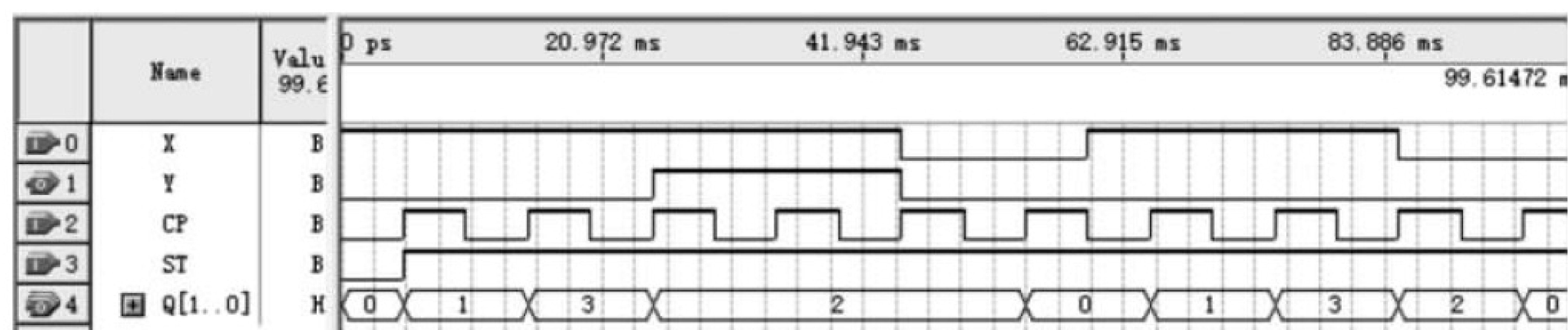


图 6.2.7　例 6.2.2 的图 3

*6.2.2　异步时序电路的分析方法



异步时序逻辑电路状态变化的特征为各个触发器状态不同时发生变化。不同时发生变化的原因是部分触发器时钟输入端未与外部时钟连接,如连接了其他触发器的输出端。显然,这些未与外部时钟连接的触发器不会随着外部时钟的到来同步发生变化。

当然,未与外部时钟连接,而连接其他触发器的输出端的触发器,状态依然可能发生变化,如与该触发器时钟输入端连接的另一个触发器的输出端恰好产生了要求的时钟信号。可见,分析异步时序电路比分析同步时序电路复杂。

分析各触发器状态变化规律时,只有那些有时钟信号的触发器才需要用状态方程去计算新态,而没有时钟信号的触发器则保持原来的状态不变。

具体可按如下步骤进行分析。

(1) 根据给定的时序电路,写出每个触发器的驱动方程(又称为激励方程)及时钟信号方程。

(2) 将驱动方程、时钟信号方程代入相应触发器的特征方程,得到每个触发器的状态方程。

(3) 找出该时序电路相对应的状态表或者状态图,以便直观地看出该时序电路的逻辑功能。

(4) 若电路中存在着无效状态(即电路未使用的状态),则应检查电路能否自启动。

(5) 用文字叙述该时序电路的逻辑功能。

【例 6.2.3】 如图 6.2.8 所示电路中各触发器均为上升沿触发,当前状态 $Q_3Q_2Q_1=000$,请问,当时钟上升沿到来时,第 2 个触发器是否需要计算新态?

---

① 在 Quartus Ⅱ环境中仿真时,当输入信号与 CP 信号同时变化时,触发器状态一般按输入信号变化前的状态变化,下同。

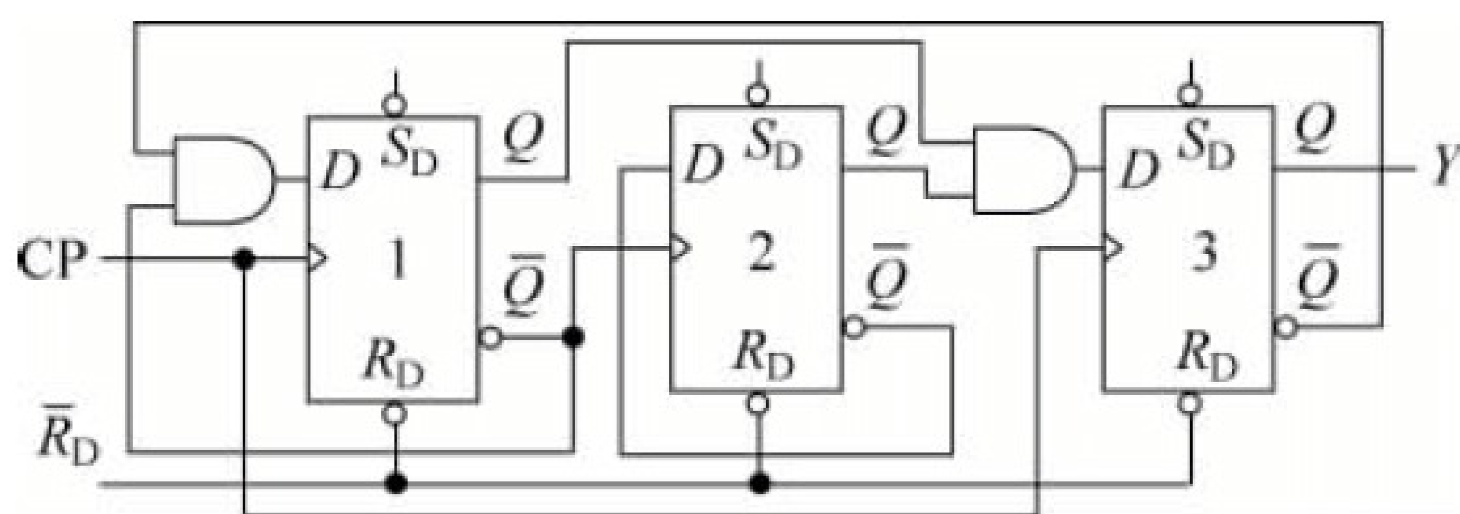


图 6.2.8 例 6.2.3 和例 6.2.4 的图

解 题中只要求答出是否需要计算新态,因此,只需研究当前状态“000”下,当时钟上升沿到来时,第 2 个触发器时钟输入端是否具有要求的上升沿信号即可求解。

(1) 先写出第 2 个触发器的时钟方程:

$$CP_2 = \overline{Q}_1^n \uparrow = Q_1 \downarrow$$

(2) 该时钟方程涉及第 1 个触发器的输出,需进一步写出第 1 个触发器的驱动方程及状态方程:

$$D_1 = \overline{Q}_1^n \ \overline{Q}_3^n, \quad CP_1 = CP \uparrow$$

$$Q_1^{n+1} = D_1 CP_1 = (\overline{Q}_1^n \ \overline{Q}_3^n) CP_1 \uparrow = (\overline{Q}_1^n \ \overline{Q}_3^n) CP \uparrow$$

(3) 得出结论。CP_1接的是外部时钟,需要计算新态,将初始状态“000”代入上面的状态方程,有

$$Q_1^{n+1} = \overline{0}\ \overline{0} = 1$$

Q_1由 0 变到 1,输出↑,无要求的时钟信号,无须计算新态。

【例 6.2.4】 试分析如图 6.2.8 所示电路的逻辑功能。

解 (1) 这是一个由三个维持阻塞 D 触发器组成的异步时序电路。它是一个 Moore 型的时序电路。因为该电路无输入信号作用。

(2) 写出电路的驱动方程、时钟方程、输出方程及状态方程。驱动方程、时钟方程如下:

$$\begin{cases} D_1 = \overline{Q}_1^n \ \overline{Q}_3^n, \quad D_2 = \overline{Q}_2^n, \quad D_3 = Q_1^n Q_2^n & (6.2.7) \\ CP_1 = CP \uparrow, \quad CP_2 = \overline{Q}_1^n \uparrow CP_3 = CP \uparrow & (6.2.8) \end{cases}$$

输出方程为

$$Y = Q_3^n \qquad (6.2.9)$$

将式(6.2.7)和式(6.2.8)代入维持阻塞 D 触发器的特性方程 $Q_i^{n+1} = (D_i) CP \uparrow$ 中,可得到电路的状态方程为

$$\begin{cases} Q_1^{n+1} = (\overline{Q}_1^n \ \overline{Q}_3^n) CP_1 \uparrow = (\overline{Q}_1^n \ \overline{Q}_3^n) CP \uparrow \\ Q_2^{n+1} = (\overline{Q}_2^n) CP_2 \uparrow = (\overline{Q}_2^n) \ \overline{Q}_1^n \uparrow \\ Q_3^{n+1} = (Q_1^n Q_2^n) CP_3 \uparrow = (Q_1^n Q_2^n) CP \uparrow \end{cases} \qquad (6.2.10)$$

(3) 画出电路的状态图。由状态方程可得到如图 6.2.9 所示的状态图。

具体求解时,依次假设电路的初始状态 $Q_3^n Q_2^n Q_1^n$,代入状态方程(式(6.2.9)),求出电路的新状态。

求解电路的新状态时,应注意每一个方程式有效的时钟条件。只有具备了时钟条件,

触发器才会按照方程式的规定更新状态，否则触发器保持原来的状态不变。需指出的是，式(6.2.8)中，CP↑是指当CP脉冲的上升沿到来时，方程才成立；否则，方程不成立，触发器维持原状态不变，类似 $\bar{Q}_1^n$↑。

例如，在状态图(图6.2.9)中，设电路的原状态为“010”，当CP脉冲的上升沿到来时(CP↑)，方程 Q_1^{n+1}、Q_3^{n+1} 成立，$Q_1^{n+1}=\bar{Q}_1^n\bar{Q}_3^n=\bar{0}\cdot\bar{0}=1$，即触发器1状态由0跳变到1，所以 $\bar{Q}_1^n$ 由1跳变到0，为下降沿，故方程 Q_2^{n+1} 不成立，触发器2维持原状态不变；又 $Q_3^{n+1}=Q_1^nQ_2^n=0\cdot1=0$，故电路的新状态为“011”。余下请读者自行分析。

(4) 检查自启动。设电路的初始状态为“101”，当CP控制脉冲到来时，将初始状态代入状态方程、输出方程，可求出输出为“1”，新状态为“010”；类似可得出电路的初始状态为“110”时，在CP脉冲作用下输出为“1”，新状态为“010”；电路的初始状态为“111”，在CP脉冲作用下输出为“1”，新状态为“100”。所以，电路能够自启动。故可以画出如图6.2.10所示的完整的状态图。

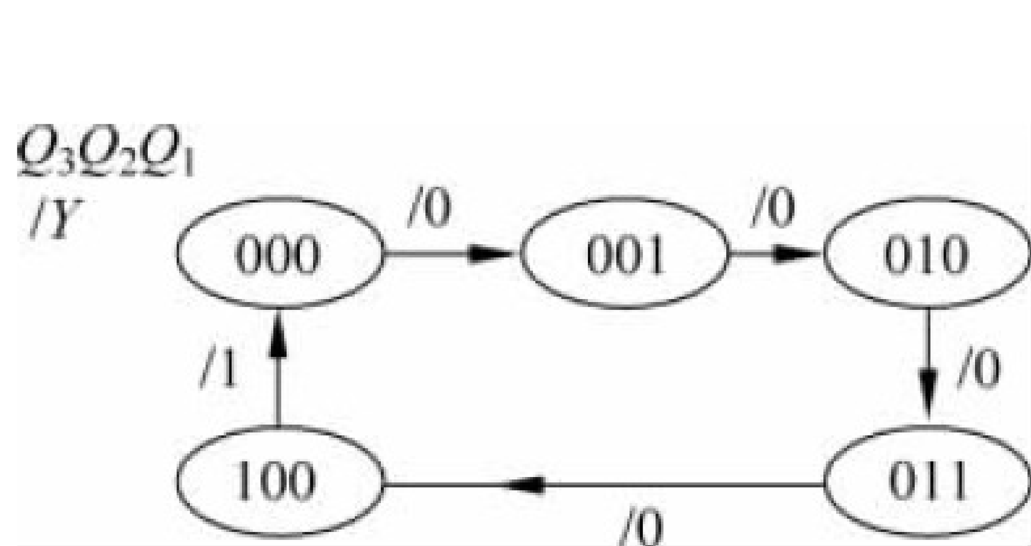


图6.2.9　例6.2.4的图1

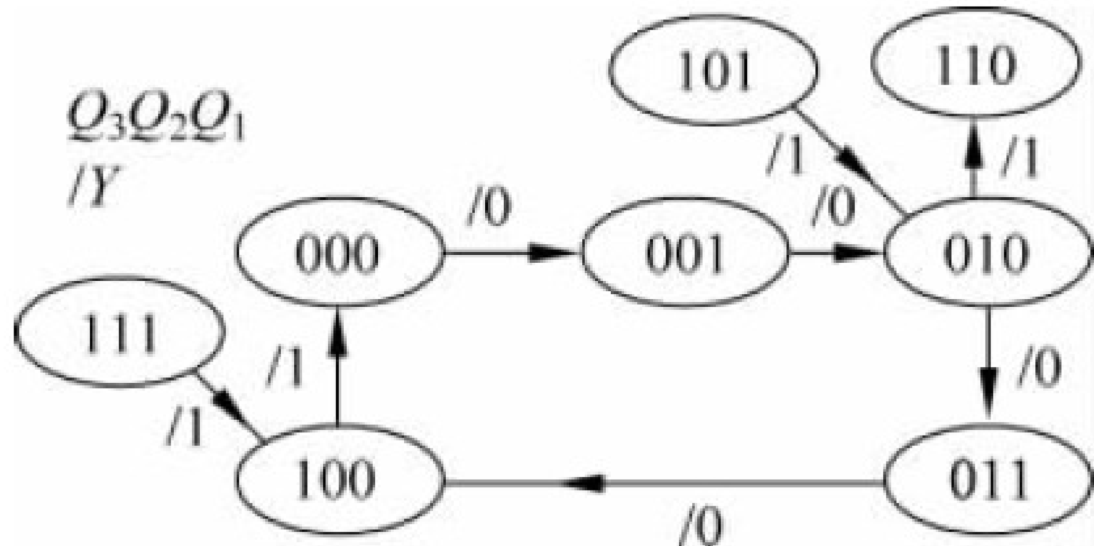


图6.2.10　例6.2.4的图2

显然，在图6.2.9中电路由无效状态转换到有效状态过程中的输出 $Y=1$，为无效输出。

(5) 结论。由如图6.2.10所示的状态图可看出，每经过5个时钟脉冲作用输出端Y输出一个进位脉冲，所以，如图6.2.8所示电路是一个能够自启动的五进制加法计数器。

(6) 计算机仿真。图6.2.8所示电路在Quartus Ⅱ环境中的仿真结果如图6.2.11所示。

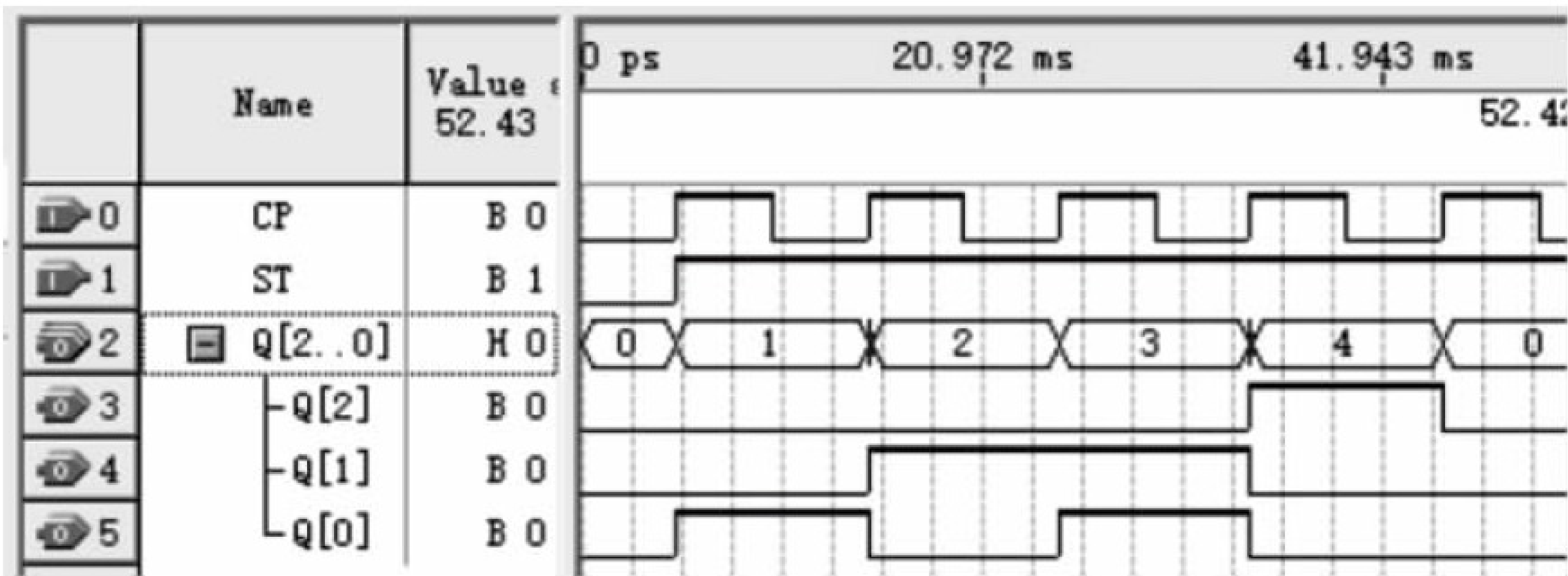


图6.2.11　例6.2.4的图3

由如图6.2.11所示仿真图可看出，图6.2.8所示电路为异步五进制上升沿计数满幅输出上升沿溢出输出下降沿的加法计数器。

将如图 6.2.11 所示仿真图中状态 1 到状态 2 之间的模糊处放大,效果如图 6.2.12 所示。

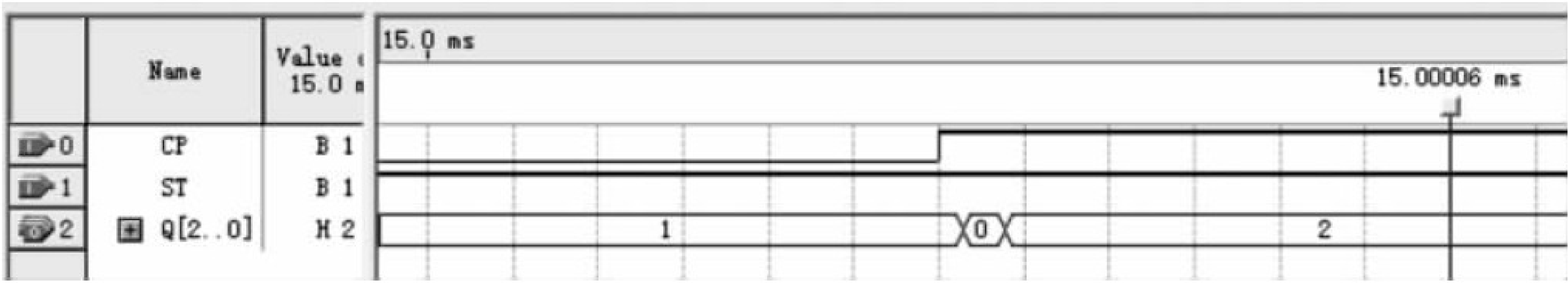


图 6.2.12　例 6.2.4 的图 4

从图 6.1.12 可看出,异步时序逻辑电路各触发器状态并不是同时发生变化,这便是异步时序电路的动作特点。



可微信扫描右边二维码学习另一个实例。

复习与思考

6.2.1　试扼要叙述同步、异步时序逻辑电路分析方法的异同。

6.2.2　试结合式(6.2.10)分析如图 6.2.8 所示电路中由状态 1 到状态 2 之间为什么会经历中间状态 0。

6.3　寄存器

寄存器是常用的时序逻辑电路之一,理解寄存器的电路构成及其特点是使用集成寄存器电路芯片的基础。



6.3.1　寄存器

1. 寄存器的含义

能够存放数码或者二进制逻辑信号的电路称为寄存器。寄存器电路由具有存储功能的触发器组成的。可通过如图 6.3.1 所示的 4 位基本寄存器来理解。

如图 6.3.1 所示电路由 4 个维持阻塞 D 触发器组成。图中,D_3、D_2、D_1、D_0 为寄存器的数据输入端,Q_3、Q_2、Q_1、Q_0 为寄存器的输出端,G 为寄存器的控制端。

当 G 上升沿到来时,依照 D 触发器的逻辑功能,有 $Q_3=D_3$,$Q_2=D_2$,$Q_1=D_1$,$Q_0=D_0$。即将 4 位二进制数写入寄存器。其他时间,依照 D 触发器的逻辑功能,触发器状态不变,即寄存器锁定原始数据不变。

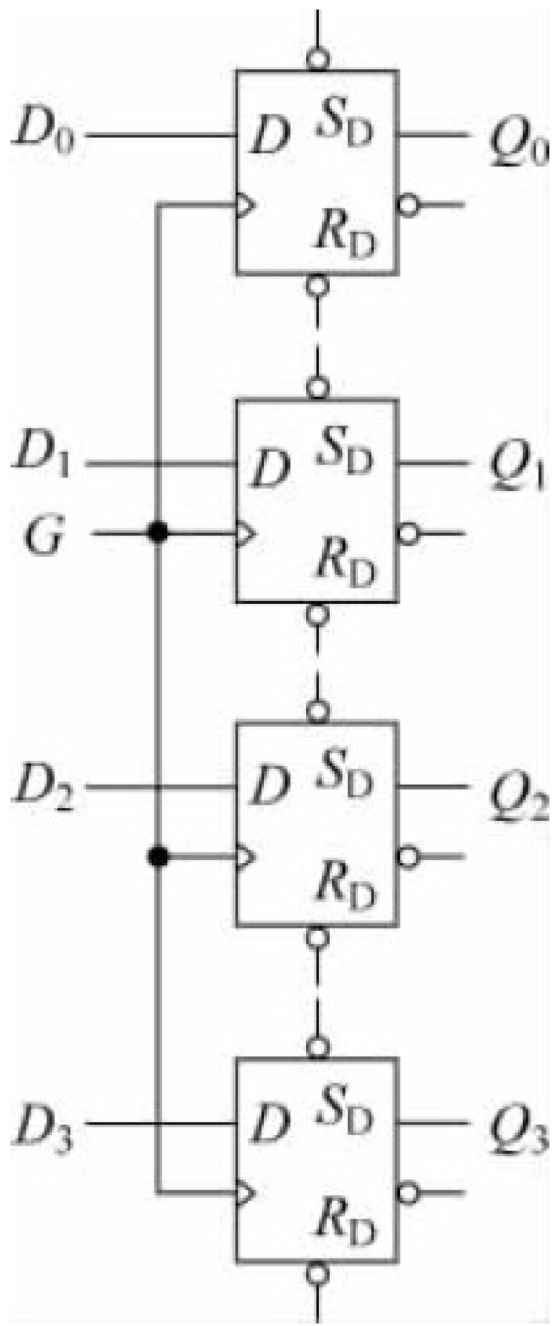


图 6.3.1　4 位基本寄存器

可见,寄存器电路是由具有存储功能的触发器组成的,用 n 个触发器组成的寄存器能存放一个 n 位的二值代码。如

图 6.3.1 所示电路由 4 个触发器组成，实现了 4 位二进制数据的存储，为 4 位的二进制数据寄存器。

2. 寄存器的种类

如图 6.3.1 所示电路所需存放的数据只能并行送入寄存器中，需要读取数据时也只能并行取出，把这种只具有并行送入并行取出存取功能的寄存器称为基本寄存器(也称并行寄存器)。

还有一类寄存器具有移位功能，称为移位寄存器。

移位寄存器不仅能够存放数据或代码，还具有移位的功能。在时钟脉冲的作用下，数据能依次逐位向左或者向右移动。

按照寄存器所存放的数据存入、取出的方式不同，移位寄存器可分为四种类型：串入-串出移位寄存器、串入-并出移位寄存器、并入-并出移位寄存器、并入-串出移位寄存器。

按照数据移动方式的不同，移位寄存器可分为三种类型：左移移位寄存器、右移移位寄存器、双向移位寄存器。

当然，不管何种类型的移位寄存器，均支持移位功能。移位功能是移位寄存器的核心功能。

3. 集成基本寄存器

为了增加使用的灵活性，在集成寄存器中，往往还增加一些控制电路，如输出三态控制。将图 6.3.1 所示电路的每一个输出端增加一个三态传输门便构成一个 4 位的输出三态寄存器。如图 6.3.2 所示为输出三态 4 位基本寄存器。当$\overline{OE}$为高电平时，寄存器输出为高阻态；当$\overline{OE}$为低电平时，寄存器正常工作。

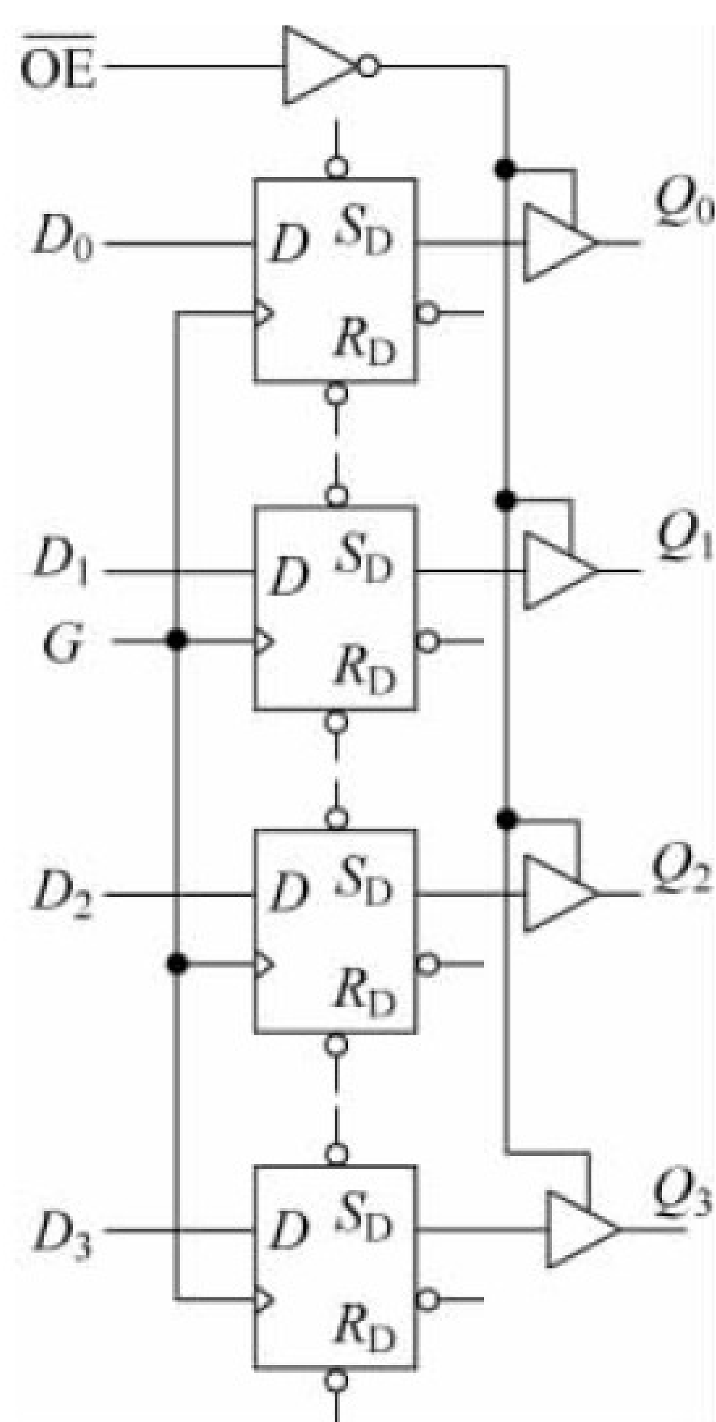


图 6.3.2　输出三态基本寄存器

6.3.2　移位寄存器的电路特点及逻辑功能



具有移位功能的寄存器称为移位寄存器。移位寄存器不但可以用来寄存数据或者代码，而且还可以用来实现数据的串行/并行的相互转换、数值的运算以及数据处理等。所以，在数字计算机中，广泛应用移位寄存器。

不管何种类型的移位寄存器，均具有数据移位功能。可通过如图 6.3.3 所示的串入-并出工作方式的 4 位移位寄存器来理解移位寄存器的移位功能。

电路中所采用的是 4 个维持阻塞 D 触发器(TTL 系列的触发器)，电路的特点为“4 个维持阻塞 D 触发器从左到右依次串接，最左边触发器接外部输入”。即从低一位触发器

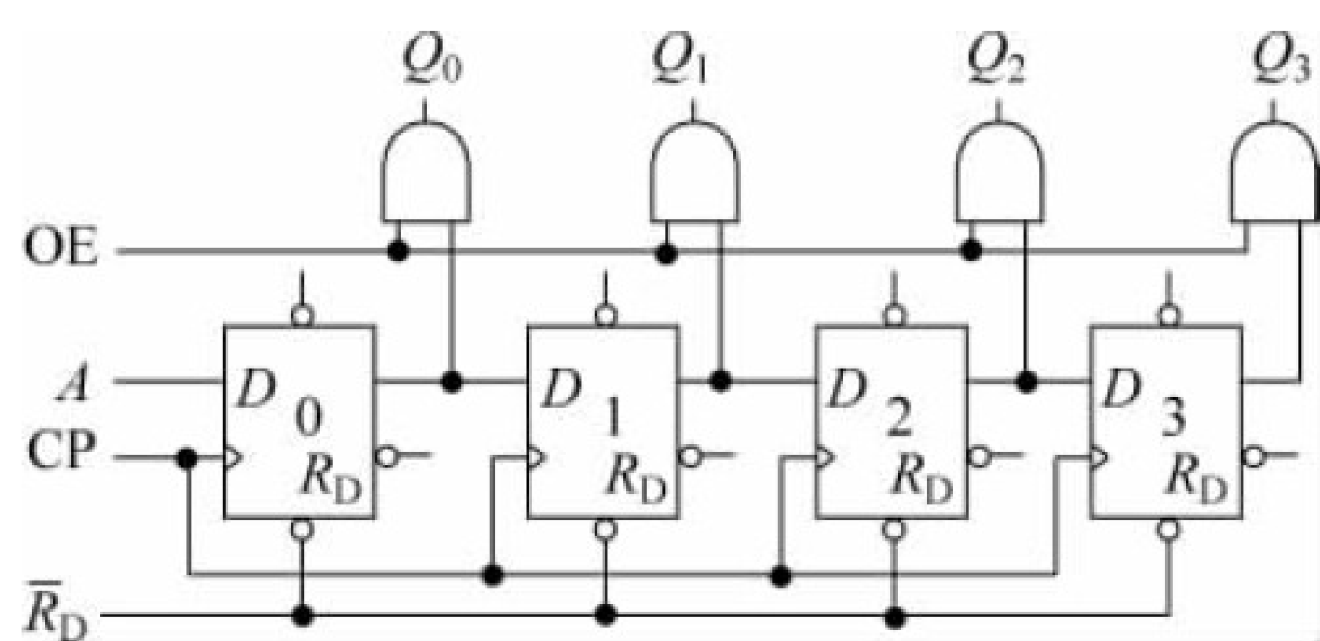


图 6.3.3　串入-并出工作方式的 4 位移位寄存器

的输出端 Q 端接到高一位触发器的输入端 D 端，最低位触发器的输入端 D 端，输入需要存放的 4 位代码。当需要将移位寄存器所存放的代码取出时，可分别通过 4 个“与”门输出。

如图 6.3.3 所示的移位寄存器是一个在 CP 控制脉冲的作用下，向右依次逐位移位的右移移位寄存器。因为所有数据只能从一个输入端输入，因此，应要求输入的代码，高位先输入，低位后输入。对数据 A，应按 A_3、A_2、A_1、A_0 的先后顺序输入数据。下面具体分析电路的工作过程。

1. 清零

在每个触发器的 $\overline{R}_D$ 端加清零负脉冲，将各触发器置“0”，此时，寄存器的状态为“0000”。

2. 放数

已知需要存放的数码为“$A_3A_2A_1A_0$”。

(1) 先送入最高位数码 A_3，第 1 个 CP↑到来时，因为

$$D_0 = A_3,\quad D_3 = D_2 = D_1 = 0$$

所以

$$Q_0^{n+1} = A_3,\quad Q_3^{n+1} = Q_2^{n+1} = Q_1^{n+1} = 0$$

寄存器的状态为“$000A_3$”($Q_3Q_2Q_1Q_0$顺序)。

(2) 先送入次高位数码 A_2，第 2 个 CP↑到来时，因为

$$D_0 = A_2,\quad D_1 = A_3,\quad D_3 = D_2 = 0$$

所以

$$Q_0^{n+1} = A_2,\quad Q_1^{n+1} = A_3,\quad Q_3^{n+1} = Q_2^{n+1} = 0$$

寄存器的状态为“$00A_3A_2$”。

(3) 继续送入低位数码 A_1，第 3 个 CP↑到来时，因为

$$D_0 = A_1,\quad D_1 = A_2,\quad D_2 = A_3,\quad D_3 = 0$$

所以

$$Q_0^{n+1}=A_1,\quad Q_1^{n+1}=A_2,\quad Q_2^{n+1}=A_3,\quad Q_3^{n+1}=0$$

寄存器的状态为“$0A_3A_2A_1$”。

(4) 送入最低位数码 A_0,第 4 个 CP↑到来时,因为

$$D_0=A_0,\quad D_1=A_1,\quad D_2=A_2,\quad D_3=A_3$$

所以

$$Q_0^{n+1}=A_0,\quad Q_1^{n+1}=A_1,\quad Q_2^{n+1}=A_2,\quad Q_3^{n+1}=A_3$$

寄存器的状态为“$A_3A_2A_1A_0$”。

故经过 4 个 CP 控制脉冲后完成了数码的存放工作,可通过图 6.3.4 来进一步理解上面的移位过程。

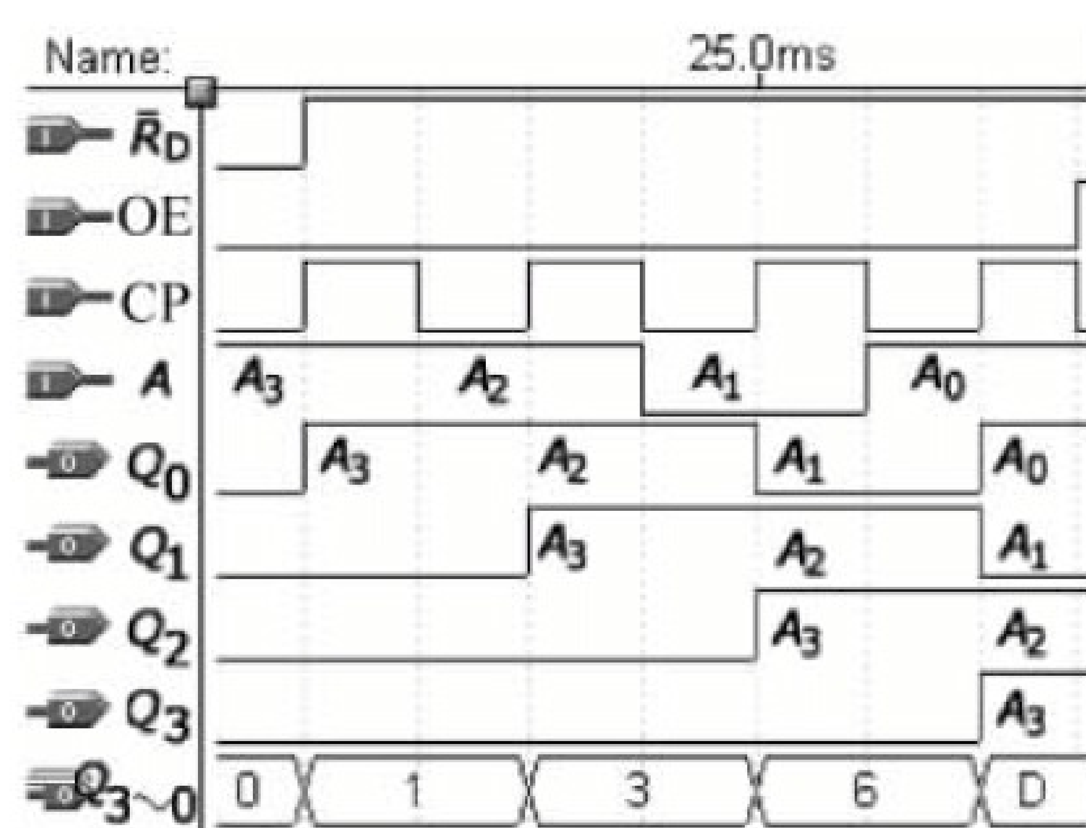


图 6.3.4 图 6.3.3 仿真波形

当如图 6.3.3 所示的移位寄存器的外部输入 D 直接悬空时(为 1),初始置 0 后,4 个时钟寄存器状态变化如下:

第 1 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1000$

第 2 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1100$

第 3 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1110$

第 4 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1111$

可总结移位寄存器右移功能的通用逻辑描述为

$$Q_i=Q_{i-1}^n \tag{6.3.1}$$

根据右移功能的通用逻辑描述,设寄存器 $Q_3Q_2Q_1Q_0$ 初始状态为“0000”,如果用右移功能设置寄存器 $Q_3Q_2Q_1Q_0$ 状态为“0101”,可从高到低,逐位移位实现即可,4 个时钟寄存器状态变化如下:

第 1 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=0000$

第 2 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1000$

第 3 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=0100$

第 4 个时钟:$Q_0Q_1Q_2Q_3=DQ_0^nQ_1^nQ_2^n=1010$

3. 取数

当读出脉冲(正脉冲)到来时,将 4 个“与”门打开,则可以同时取出在移位寄存器中所存放的数码“A_3、A_2、A_1、A_0”。

从对移位寄存器的工作过程的分析中,可以看出,存在着空翻现象的触发器不能组成移位寄存器。

可类推总结移位寄存器左移通用逻辑描述为

$$Q_i = Q^n_{(i+1)} \tag{6.3.2}$$

参考如图 6.3.3 所示电路,可类推总结左移电路的连接特点为“4 个维持阻塞 D 触发器从右到左依次串接,最右边触发器接外部输入”,参考电路如图 6.3.5 所示。

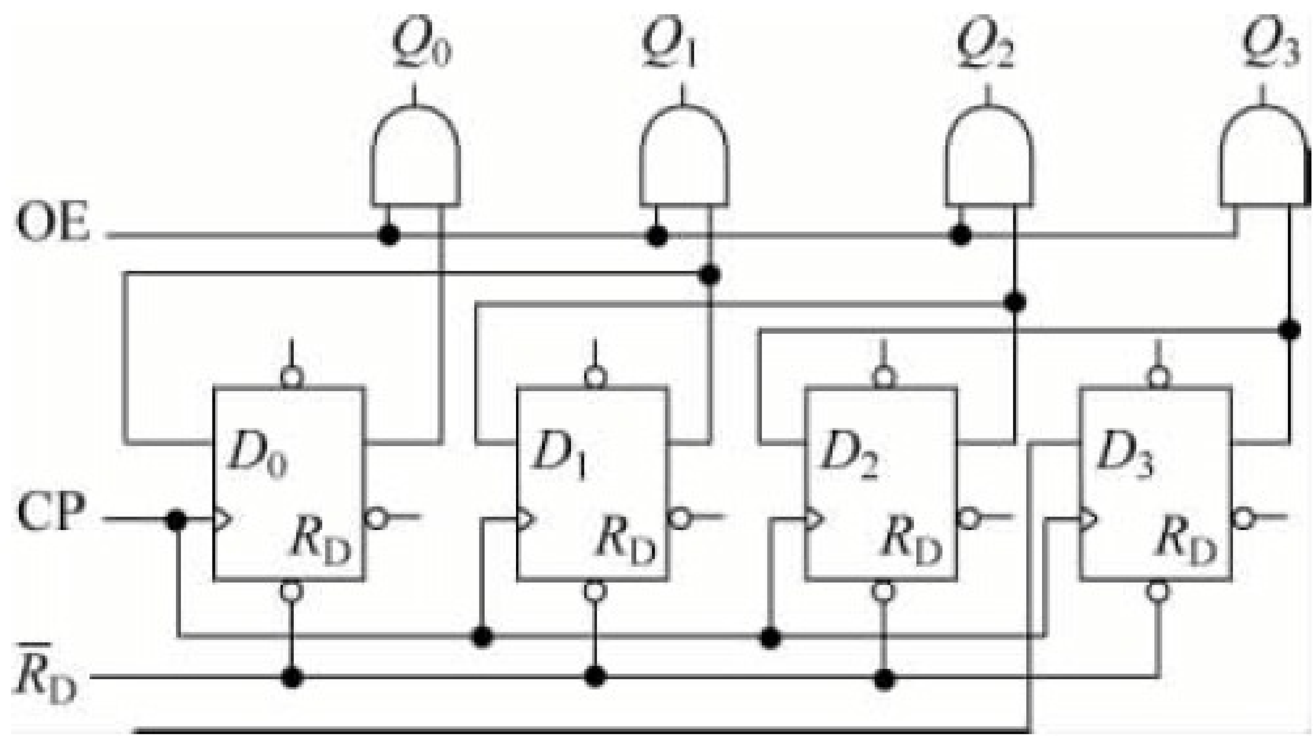


图 6.3.5 4 位左移移位寄存器

还有一种寄存器既可以左移,又可以右移,称为双向移位寄存器。读者可对照图 6.3.3 和图 6.3.5 总结双向移位寄存器的构成方法。

当然,移位寄存器还常用 JK 触发器构成。读者可参考例 5.3.1 介绍的 JK 触发器到 D 触发器的转换方法,对照图 6.3.3 和图 6.3.5 设计用 JK 触发器实现的移位寄存器电路。

6.3.3 集成移位寄存器的逻辑功能

在移位寄存器的基础上,增加了一些辅助功能(如清零、置数、保持等)便构成集成移位寄存器。集成移位寄存器的主要产品有:4 位移位寄存器 74LS195、4 位双向移位寄存器 74LS194;8 位移位寄存器 74LS164、8 位双向移位寄存器 74LS198 等。

1. 集成单向移位寄存器

(1) 引脚说明。如图 6.3.6 所示为 4 位移位寄存器 74LS195。图中,$\overline{CR}$是清零端,SH/$\overline{LD}$是移位置数控制端,$D_0 \sim D_3$是并行数据输入端,J、$\overline{K}$ 为数据输入端;$Q_0 \sim Q_3$是寄存器输出端。

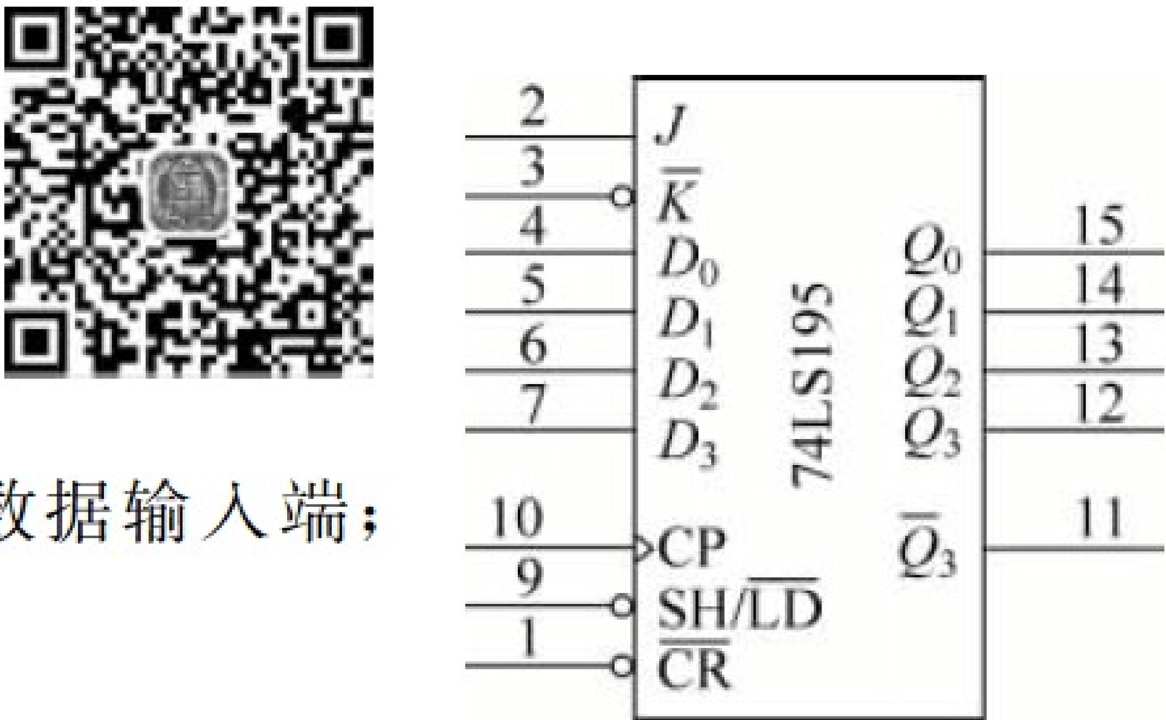


图 6.3.6 74LS195

(2) 功能表。74LS195 功能表如表 6.3.1 所示。由表可看出,集成 4 位移位寄存器 74LS195 具有如下功能。

表 6.3.1　74LS195 功能表

$\overline{CR}$	SH/$\overline{LD}$	J$\overline{K}$	CP	D_0 D_1 D_2 D_3	Q_0 Q_1 Q_2 Q_3
0	×	××	×	× × × ×	0 0 0 0
1	0	××	↑	d_0 d_1 d_2 d_3	d_0 d_1 d_2 d_3
1	1	0 1	↑	× × × ×	Q_0 Q_0 Q_1 Q_2
1	1	0 0	↑	× × × ×	0 Q_0 Q_1 Q_2
1	1	1 0	↑	× × × ×	$\overline{Q}_0$ Q_0 Q_1 Q_2
1	1	1 1	↑	× × × ×	1 Q_0 Q_1 Q_2
1	1	××	0	× × × ×	Q_0 Q_1 Q_2 Q_3

① 清零功能。功能表第 1 行表明，$\overline{CR}$为清零输入引脚。当$\overline{CR}=0$ 时，移位寄存器异步清零。

② 并行送数功能。功能表第 2 行表明，SH/$\overline{LD}$=0 时，为并行置数控制引脚。当$\overline{CR}=1$，SH/$\overline{LD}$=0 时，在 CP 上升沿的作用下，可将加在并行输入端 $D_0 \sim D_3$的数码 $d_0 \sim d_3$送入移位寄存器中。

③ 右移串行送数功能。功能表第 3～6 行表明，SH/$\overline{LD}$=1 时，为右移移位控制引脚。当 $\overline{CR}=1$，SH/$\overline{LD}$=1 时，在 CP 上升沿的作用下，执行右移位寄存器功能，Q_0接收 J、$\overline{K}$ 串行输入数据。

根据右移逻辑功能特点，有

$$Q_1Q_2Q_3 = Q_0^nQ_1^nQ_2^n$$

Q_0的值由 JK 触发器外部输入 J、$\overline{K}$ 确定。

$$Q_0 = J\overline{Q}_0 + \overline{K}Q_0^n \qquad (6.3.3)$$

根据式(6.3.3)，如果将外部输入 J、$\overline{K}$ 接外部输入 T，有

$$J = \overline{K} = T, \quad Q_0 = T\overline{Q}_0^n + TQ_0^n = T$$

④ 保持功能。功能表第 7 行表明，当$\overline{CR}=1$，SH/$\overline{LD}$=1，CP=0 时，移位寄存器保持状态不变。

74LS195 的逻辑功能总结为：74LS195 为异步清零、同步置数的右移移位寄存器。Q_0的值依照外部输入 J、$\overline{K}$ 的值按照 JK 功能触发器发生变化。

当然，电路的逻辑功能是由电路的结构决定的，Quartus Ⅱ 中 74LS195 的电路原理图如图 6.3.7 所示。其中，CLRN 对应$\overline{CR}$，ST/LDN 对应 SH/$\overline{LD}$，KN 对应 $\overline{K}$，CLK 对应 CP。

如当并行置数无效时，外部输入数据 D_0被屏蔽，有

$$Q_0 = D = J\overline{Q}_0^n + KNQ_0^n$$

有兴趣的读者可参考该电路进一步理解 74LS195 的逻辑功能。

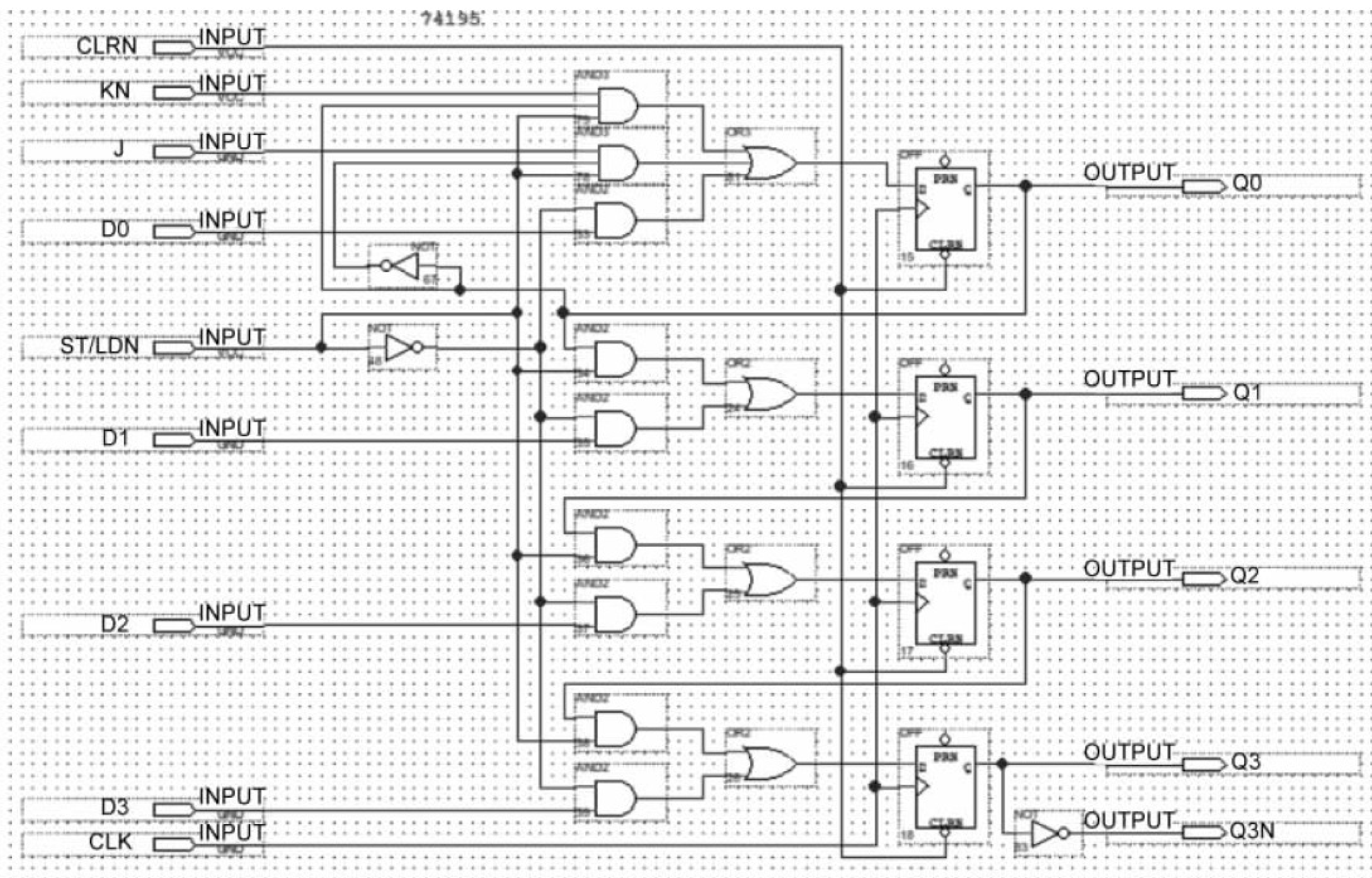


图 6.3.7　74LS195 的电路原理图

2. 集成双向移位寄存器



(1) 引脚说明。如图 6.3.8 所示为 4 位移位寄存器 74LS194。图中，$\overline{CR}$为清零端，M_0、M_1为工作方式控制端，D_{SL}为左移串行数据输入端，D_{SR}为右移串行数据输入端，D_0～D_3为并行数据输入端；Q_0～Q_3为寄存器输出端。

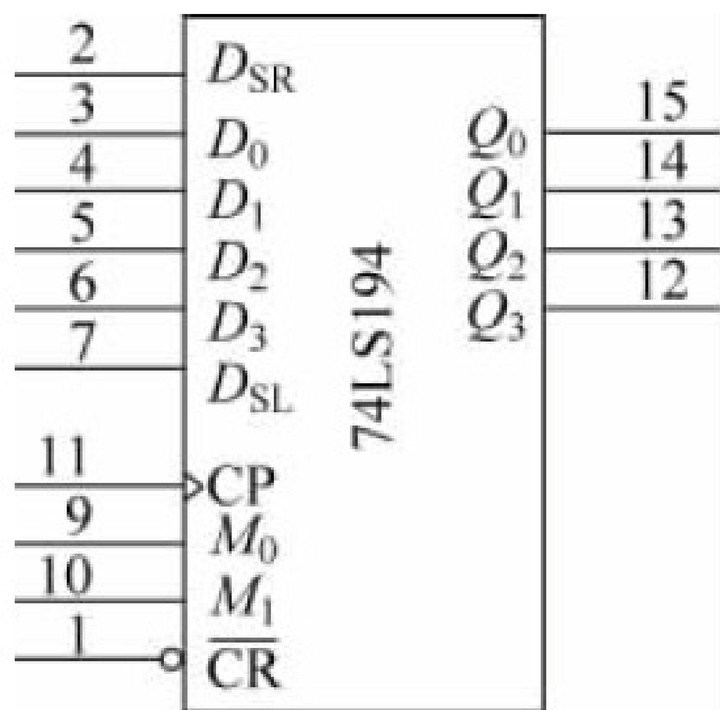


图 6.3.8　74LS194

(2) 功能表。74LS194 功能表如表 6.3.2 所示。由表可看出，集成 4 位移位寄存器 74LS194 具有如下功能。

表 6.3.2　74LS194 功能表

$\overline{CR}$	M_1 M_0	D_{SL}	D_{SR}	CP	D_0 D_1 D_2 D_3	Q_0 Q_1 Q_2 Q_3
0	× ×	×	×	×	× × × ×	0 0 0 0
1	× ×	×	×	0	× × × ×	Q_0 Q_1 Q_2 Q_3
1	1 1	×	×	↑	d_0 d_1 d_2 d_3	d_0 d_1 d_2 d_3
1	0 1	×	1	↑	× × × ×	1 Q_0 Q_1 Q_2
1	0 1	×	0	↑	× × × ×	0 Q_0 Q_1 Q_2
1	1 0	1	×	↑	× × × ×	Q_1 Q_2 Q_3 1
1	1 0	0	×	↑	× × × ×	Q_1 Q_2 Q_3 0
1	0 0	×	×	×	× × × ×	Q_0 Q_1 Q_2 Q_3

① 清零功能(真值表第 1 行)。当$\overline{CR}=0$时,双向移位寄存器异步清零,移位寄存器各触发器立即全部复位为 0。

② 保持功能(功能表第 2、8 行)。当$\overline{CR}=1$时,CP=0 或 $M_0=M_1=0$,双向移位寄存器保持状态不变。

③ 并行输入数据功能(功能表第 3 行)。当$\overline{CR}=1$,$M_0=M_1=1$ 时,在 CP 脉冲上升沿的作用下,可将加在并行输入端 $D_0\sim D_3$的数据 $d_0\sim d_3$送入寄存器中。

④ 右移串行输入数据功能(功能表第 4、5 行)。当$\overline{CR}=1$,$M_0=1$,$M_1=0$ 时,为右移移位寄存器,在 CP 脉冲上升沿的作用下,有

$$Q_i = Q_{i-1}^n,\quad Q_0 = D_{SR} \tag{6.3.4}$$

⑤ 左移串行输入数据功能(功能表第 6、7 行)。当$\overline{CR}=1$,$M_0=0$,$M_1=1$ 时,为左移移位寄存器,在 CP 脉冲上升沿的作用下,有

$$Q_i = Q_{i+1}^n,\quad Q_3 = D_{SL} \tag{6.3.5}$$

(3) 74LS194 的逻辑功能总结为:74LS194 为异步清零、同步置数的双向移位寄存器,具体方式主要由 M_1M_0 确定。

【例 6.3.1】 请用两片 74LS194 构成一个 8 位的双向移位寄存器。

解 (1) 8 位并行输入、输出数据的实现。定义片 1 的 $D_0\sim D_3$对应 8 位寄存器的 $D_0\sim D_3$、片 2 的 $D_0\sim D_3$对应 8 位寄存器的 $D_4\sim D_7$;定义片 1 的 $Q_0\sim Q_3$对应 8 位寄存器的$Q_0\sim Q_3$、片 2 的 $Q_0\sim Q_3$对应 8 位寄存器的 $Q_4\sim Q_7$。

(2) $\overline{CR}$、M_1、M_0、CP 4 个公共控制引脚短接。

(3) 右移功能的实现。当$\overline{CR}=1$,$M_0=1$,$M_1=0$ 时,片 1、片 2 为两个独立的右移移位寄存器,有

$$Q_i = Q_{i-1}^n,\quad Q_0 = D_{SR}$$

依据 8 位右移移位功能要求,有

$$Q_4 = Q_3^n$$

依照(1)、(2)的连接特点,有

$$Q_4 = Q_{0(2)} = D_{SR(2)},\quad Q_3 = Q_{3(1)}$$

令 $D_{SR(2)}=Q_{3(1)}$ 即可。

(4) 左移功能的实现。依据 8 位左移移位功能要求,有

$$Q_3 = Q_4^n$$

依照(1)、(2)的连接特点,有

$$Q_4 = Q_{0(2)},\quad Q_3 = D_{SL(1)}$$

令 $D_{SL(1)}=Q_{0(2)}$ 即可。

可画出电路如图 6.3.9 所示。

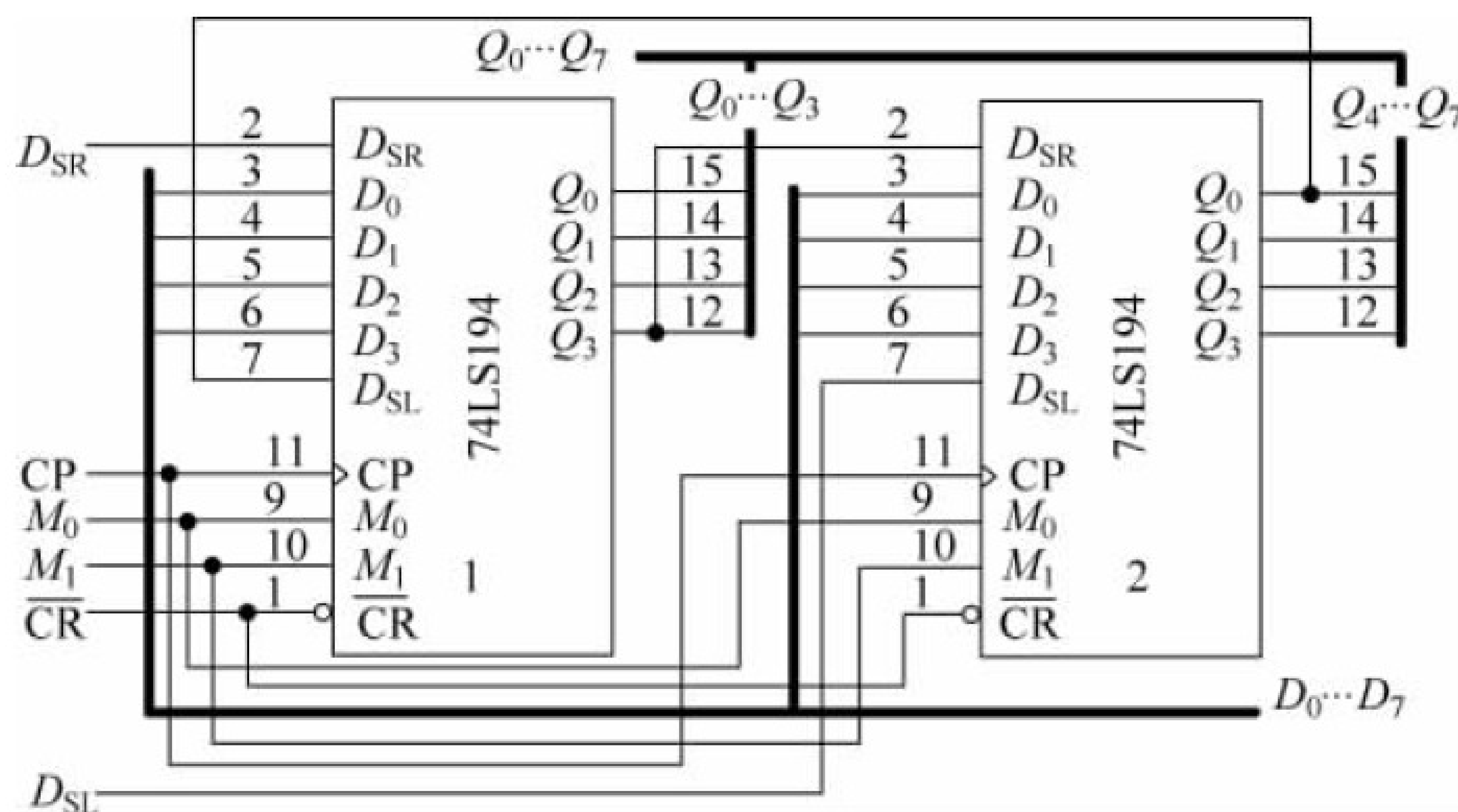


图 6.3.9　例 6.3.1 的图

参照用两片 194 构成的 8 位双向移位寄存器的连接方法，可画出用两片 195 构成的 8 位移位寄存器电路如图 6.3.10 所示。

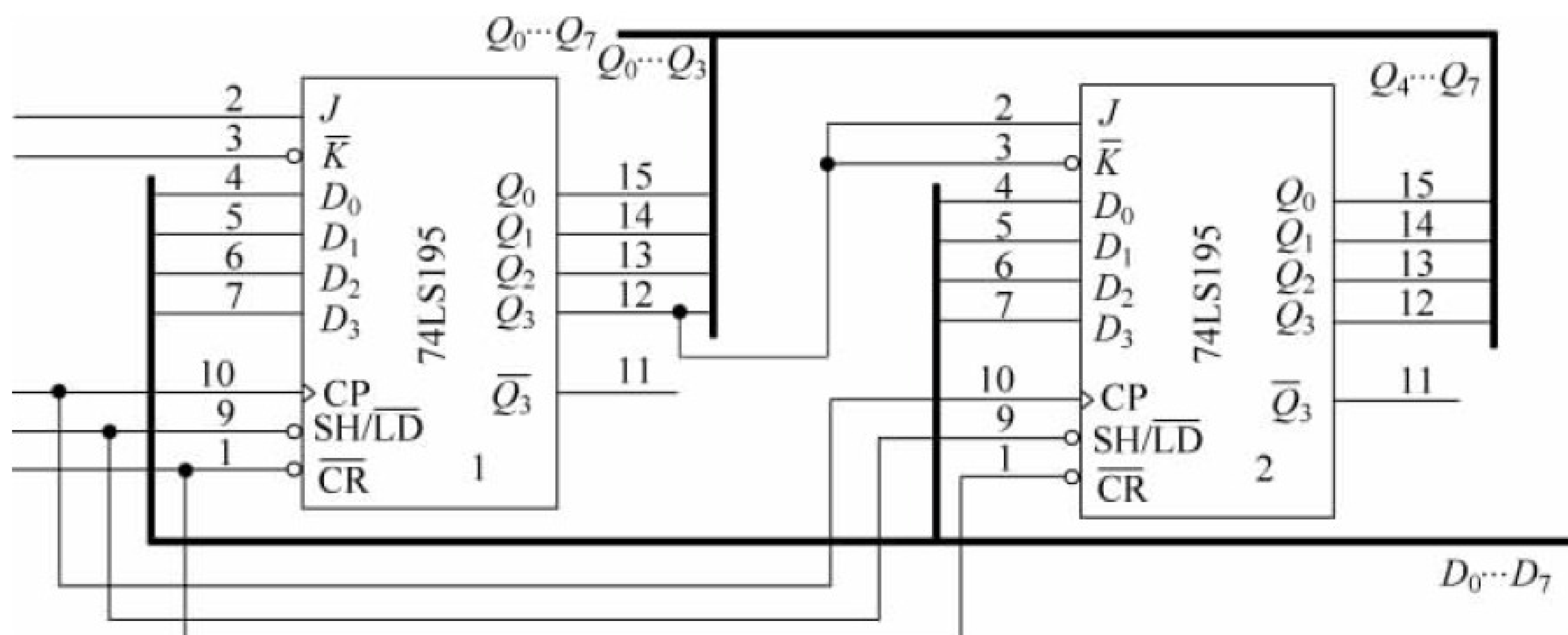


图 6.3.10　用 74LS195 构成的 8 位右移寄存器

8 位并行输入、输出数据，公共控制引脚连接方法与例 6.3.1 中介绍的方法相同。

74LS195 为右移移位寄存器，有

$$Q_4 = Q_3^n$$

根据电路的连接特点，有

$$Q_4 = Q_{0(2)} = J_2\overline{Q}_0^n + \overline{K}_2 Q_0^n$$

将片 2 的外部输入 J_2、$\overline{K}_2$ 短接后与片 1 的 Q_3 连接即可。

QuartusⅡ中 74LS194 的电路原理图如图 6.3.11 所示。其中，SRSI 对应 D_{SR}，SLSI 对应 D_{SL}，CLRN 对应 $\overline{CR}$，S_0、S_1 分别对应 M_0、M_1，KN 对应 $\overline{K}$，CLK 对应 CP，A B CD 对应 $D_0D_1D_2D_3$，QAQBQCQD 对应 $Q_0Q_1Q_2Q_3$。

如当 $S_1S_0=01$ 时，SRSI 输入有效，外部输入 A 被 S_1 屏蔽，QA^n 被 $\overline{S}_0$ 屏蔽，则有

$$QA = \text{SRSI}, \quad \text{即 } Q_0 = D_{SR}$$

有兴趣的读者可参考该电路进一步理解 74LS194 的逻辑功能。

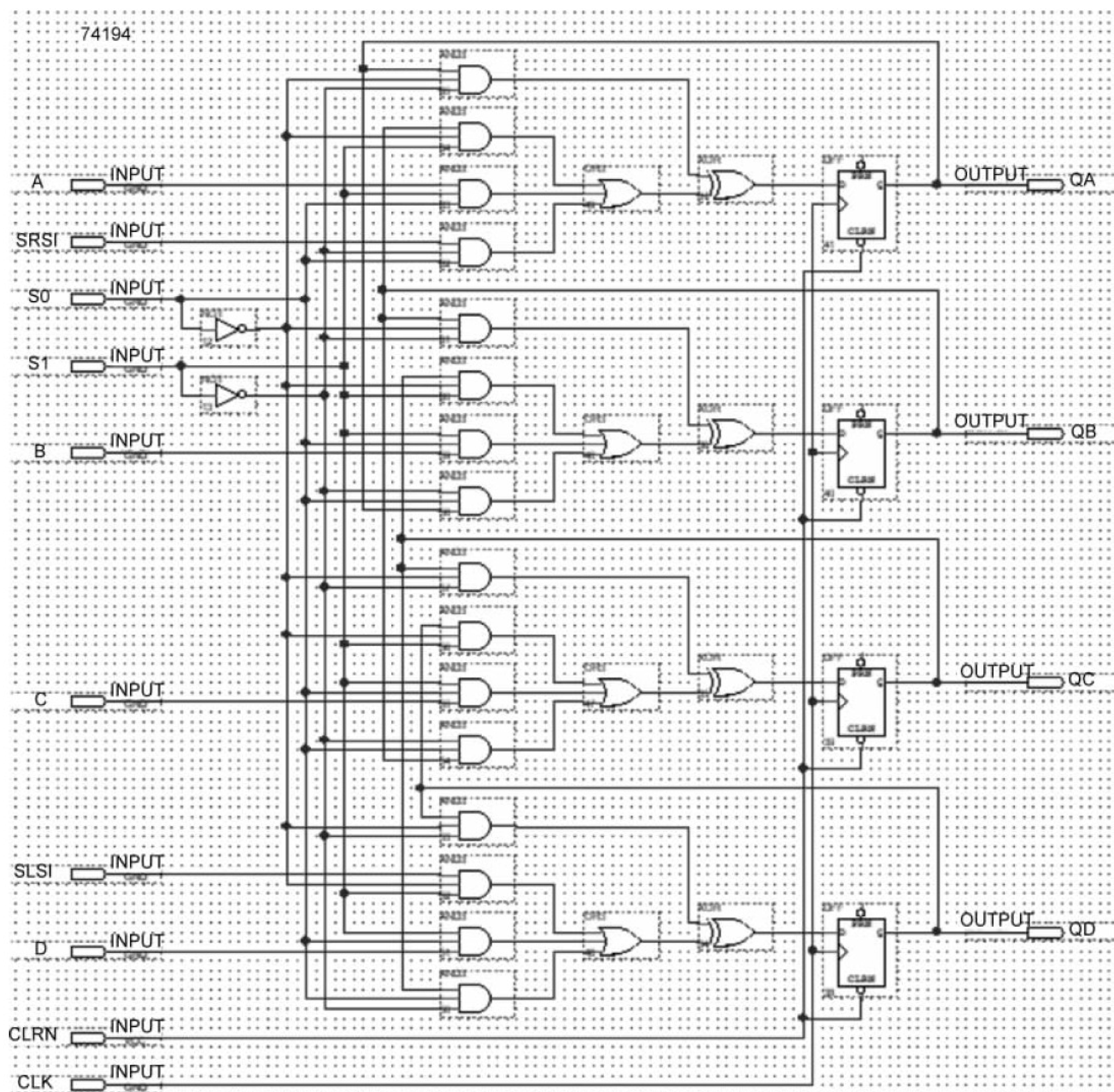


图 6.3.11 74LS194 的电路原理图

6.3.4 移位寄存器的应用

移位寄存器应用十分广泛，主要应用有如下几种。

1. 存放数据

移位寄存器首先是寄存器，存放数据是寄存器的基本功能。寄存器是计算机 CPU 中的基本部件，是理解 CPU 的基础。

2. 数学运算

二进制数据在移位寄存器中右移 1 位等于数学上左移 1 位，数值上等于乘以 2。类似

地，数据在移位寄存器中左移 1 位等于数学上右移 1 位，数值上等于除以 2。因此，移位寄存器广泛用于数学运算。

3. 实现数码的串行、并行变换

在数字通信系统中，通信线路上信息传递通常是串行传送，而终端的输入或输出往往是并行的，因而需要将串行信号变换成并行信号或者由并行信号变换成串行信号。例如，图 6.3.3 所示 4 位移位寄存器就可将串行信号变换成并行信号。如果串行输入的数码为 $A_3A_2A_1A_0=1101$，则图 6.3.3 所示电路在 Quartus Ⅱ环境中的仿真结果如图 6.3.4(经过处理)所示。从图中可以看出，移位寄存器的移存脉冲(CP 脉冲)与代码的码元应同步。并行读出脉冲(OE)必须在 4 个 CP 脉冲后出现，并且和 CP 脉冲出现的时间互相错开。

4. 移位寄存器可以作为脉冲节拍延迟

由于移位寄存器在串行输入、串行输出时，输入信号经过 n 次移位后才到达输出端输出，所以输出信号比输入信号延迟了 n 个移位脉冲的周期，故起到了节拍延迟的作用。其延迟的时间为

$$t_{\mathrm{d}}=nt_{\mathrm{CP}}$$

此外，移位寄存器还常用来构成计数器、序列信号发生器等常用时序逻辑电路。

复习与思考

6.3.1　设寄存器 $Q_3Q_2Q_1Q_0$ 初始状态为“0000”，用左移功能设置寄存器 $Q_3Q_2Q_1Q_0$ 状态为“0011”，写出完成该设置过程中的 4 个时钟周期寄存器的状态变化情况。

6.3.2　用两片 194 构成一个 8 位左移寄存器。

6.4 计数器

计数器不仅可以用来计数，而且还可以用来作为定时器、分频器、脉冲序列发生器、数字仪表以及在数字计算机中用于数字运算等，是应用最广泛的一类时序逻辑电路。

6.4.1 计数器的含义及种类



统计脉冲的个数称为计数，实现计数功能的电路称为计数器。计数器应用十分广泛，主要分类方法有：

1. 按计数器中触发器工作是否与时钟脉冲同步分类

同步计数器——输入的时钟脉冲(又称为计数脉冲)同时作用于电路中的所有触发器，这种计数器称为同步计数器。

异步计数器——输入的计数脉冲到来时，各个触发器的工作是异步进行的，这种计数器称异步计数器。从电路结构上看，计数器中的各个触发器，有的触发器其时钟信号是输入的计数脉冲，有的触发器其时钟信号却是其他触发器的输出。

2. 按计数的进制分类

二进制计数器——当输入的计数脉冲到来时，按二进制规律进行计数的计数器称为二进制计数器。

十进制计数器——按十进制规律进行计数的计数器称为十进制计数器。原理上，二进制计数器与十进制计数器并无本质区别。

N 进制计数器——除了二进制计数器和十进制计数器之外的其他进制数的计数器，都称为 N 进制计数器。

3. 按计数时是递增还是递减分类

加法计数器——当输入的计数脉冲到来时，按递增规律进行计数的计数器称为加法计数器。

减法计数器——当输入的计数脉冲到来时，按递减规律进行的计数器称为减法计数器。

可逆计数器——在加、减信号的控制下，既可以进行递增计数也可进行递减计数的计数器称为可逆计数器。

6.4.2 同步计数器的电路构成特点

同步计数器是典型的同步时序电路，电路中所有的触发器都是共用同一个时钟脉冲源，这个时钟脉冲源就是被计数的输入脉冲。

1. 同步 3 位二进制加法计数器

构成二进制计数器方法较多，这里介绍一种通用方法。同步 3 位二进制加法计数器实例如图 6.4.1 所示，该图在 Quartus Ⅱ环境中的仿真结果如图 6.4.2 所示。图中的数字输出形式为 3 个触发器按照 $Q_2Q_1Q_0$（电路中为 $Q_3Q_2Q_1$）的顺序以总线形式仿真的结果。

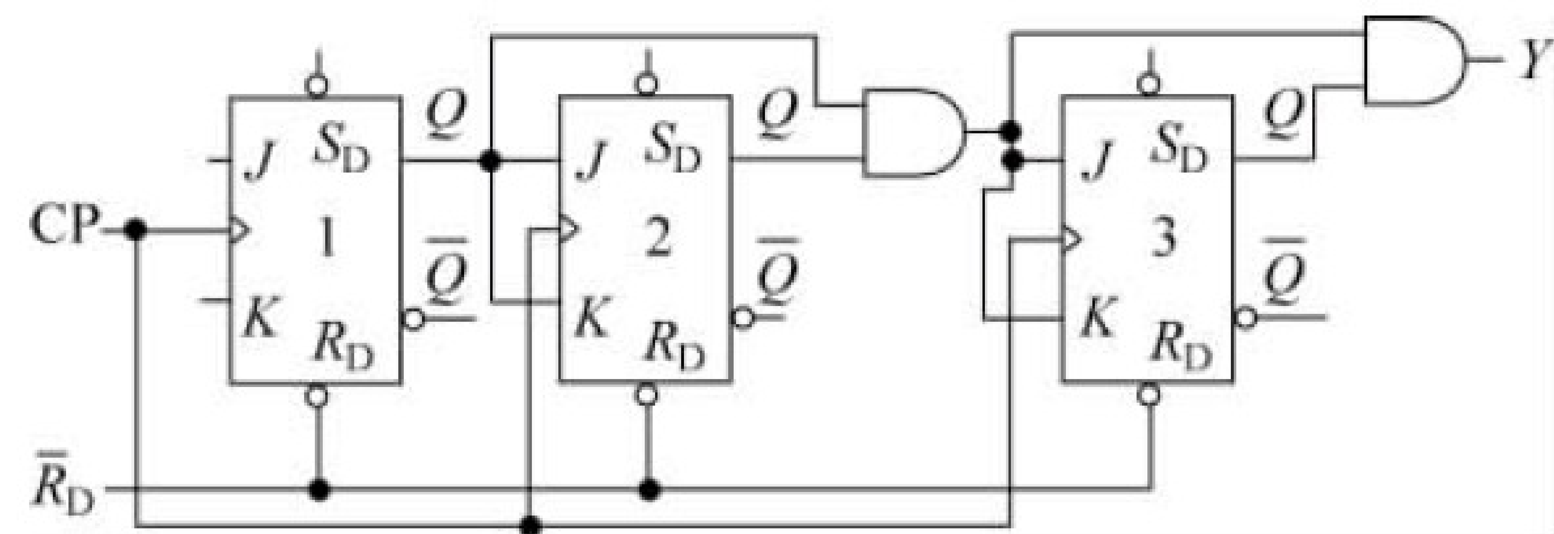


图 6.4.1 3 位二进制加法计数器的图

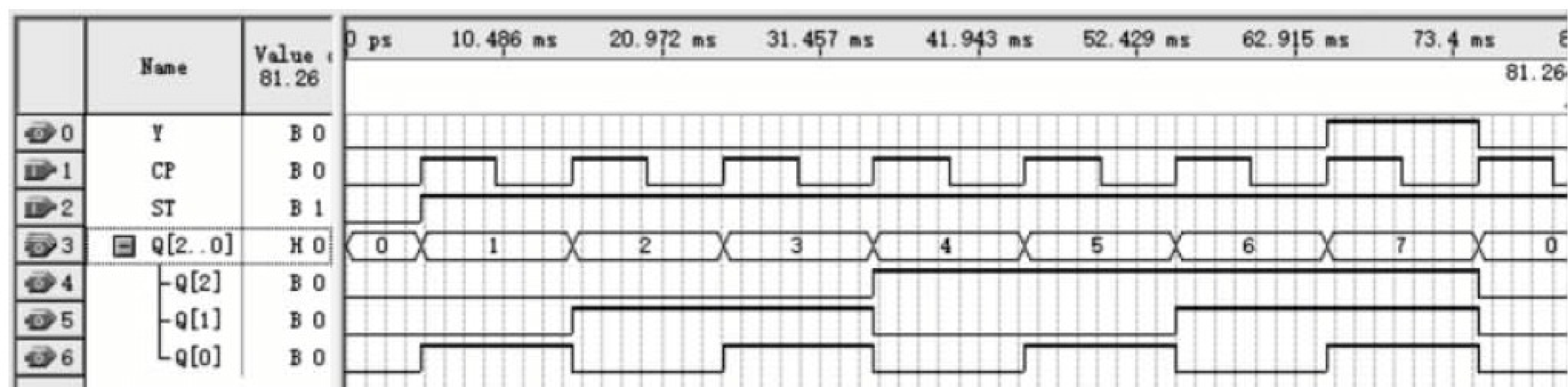


图 6.4.2　如图 6.4.1 所示电路的仿真图

由图 6.4.2 可得到如图 6.4.3 所示的状态图。从状态图、仿真图可看出，图 6.4.1 所示电路每 8 个 CP 时钟 $Q_3Q_2Q_1$ 由状态 0 回到状态 0，为对时钟信号计数的 3 位二进制加法计数器，也称八进制加法计数器。

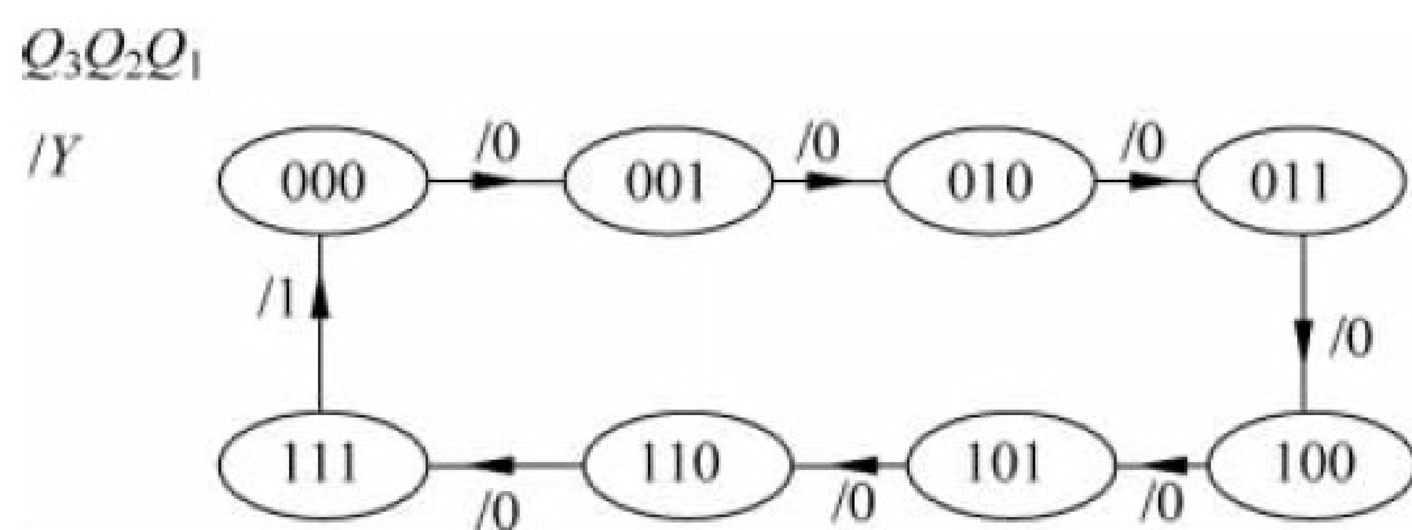


图 6.4.3　如图 6.4.1 所示电路的状态图

一个计数器所能够记入计数脉冲的数目，称为计数器的计数容量、计数长度或计数器的模。上述 3 位二进制计数器的计数容量等于 8，其计数长度或模值数也等于 8。

由仿真时序图可以看出，若输入计数脉冲的频率为 f_0，则 Q_1、Q_2、Q_3 端可以依次输出频率为 $\frac{1}{2}f_0$、$\frac{1}{4}f_0$、$\frac{1}{8}f_0$ 的周期性的矩形脉冲。因为计数器具有分频功能，所以也称为分频器。

如图 6.4.1 所示的八进制计数器，每 8 个时钟脉冲，输出 Y 输出 1 个脉冲，因此，也称八分频器。

也可根据 JK 触发器特点直接求出如图 6.4.1 所示电路的仿真时序图，求解思路如下。

如图 6.4.1 所示电路中第 1 个触发器连接特点：$J_1=K_1=1$，时钟端接外部时钟 CP。当 CP↑ 到来时，Q_1（仿真图中为 Q_0）翻转。可由 CP 波形直接画出 Q_1 的波形，如图 6.4.2 中最下一行所示。

第 2 个触发器连接特点：$J_2=K_2=Q_1^n$，时钟端接外部时钟 CP。当 CP↑ 到来且 $Q_1^n=1$ 时，Q_2（仿真图中为 Q_1）翻转，其余时间保持状态不变，可由 CP 波形及 Q_1 的波形直接画出 Q_2 的波形，如图 6.4.2 中倒数第 2 行所示（仿真图中的 Q_0 波形对应 Q_1^{n+1}，而不是 Q_1^n，下同）。

第 3 个触发器连接特点：$J_3=K_3=Q_1^nQ_2^n$，时钟端接外部时钟 CP。当 CP↑ 到来且 $Q_1^n=Q_2^n=1$ 时，Q_3（仿真图中为 Q_2）翻转，其余时间保持状态不变，可由 CP 波形及 Q_1、Q_2 的波形直接画出 Q_3 的波形，如图 6.4.2 中倒数第 3 行所示。

由如图 6.4.2 所示仿真时序图中输出 Y 及触发器各状态变化特点，可进一步总结如

图 6.4.1 所示电路的逻辑特点：同步八进制上升沿计数、满幅输出上升沿、溢出输出下降沿的加法计数器。



2. 同步 4 位二进制加法计数器

同步 4 位二进制加法计数器实例如图 6.4.4 所示，该图在 Quartus Ⅱ 环境中的仿真结果如图 6.4.5 所示。图中的数字输出形式为 4 个触发器按照 $Q_3Q_2Q_1Q_0$（电路中为 $Q_4Q_3Q_2Q_1$）的顺序以总线形式仿真的结果。

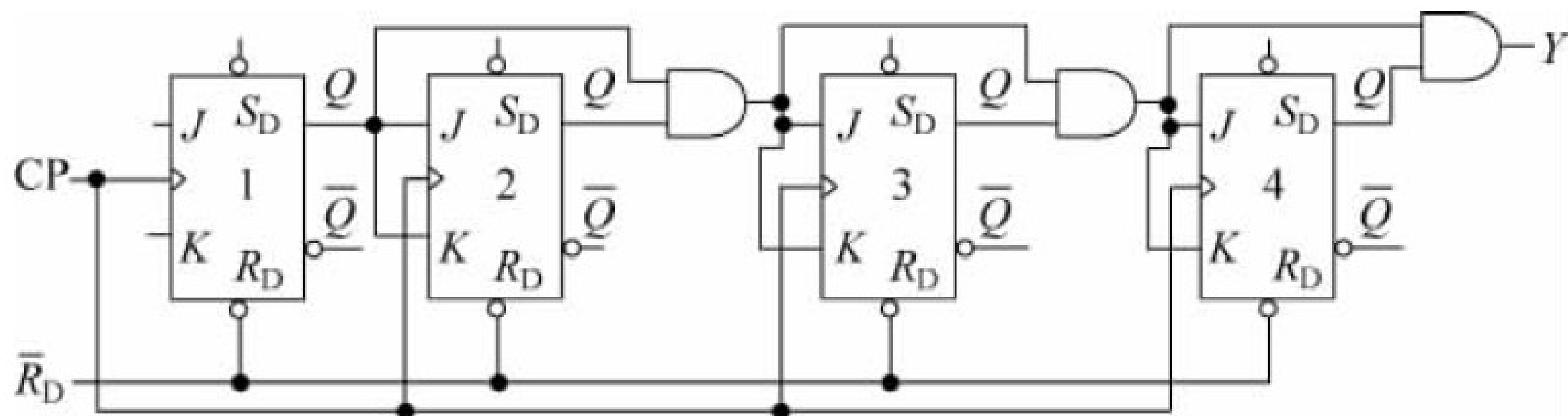


图 6.4.4　同步 4 位二进制加法计数器

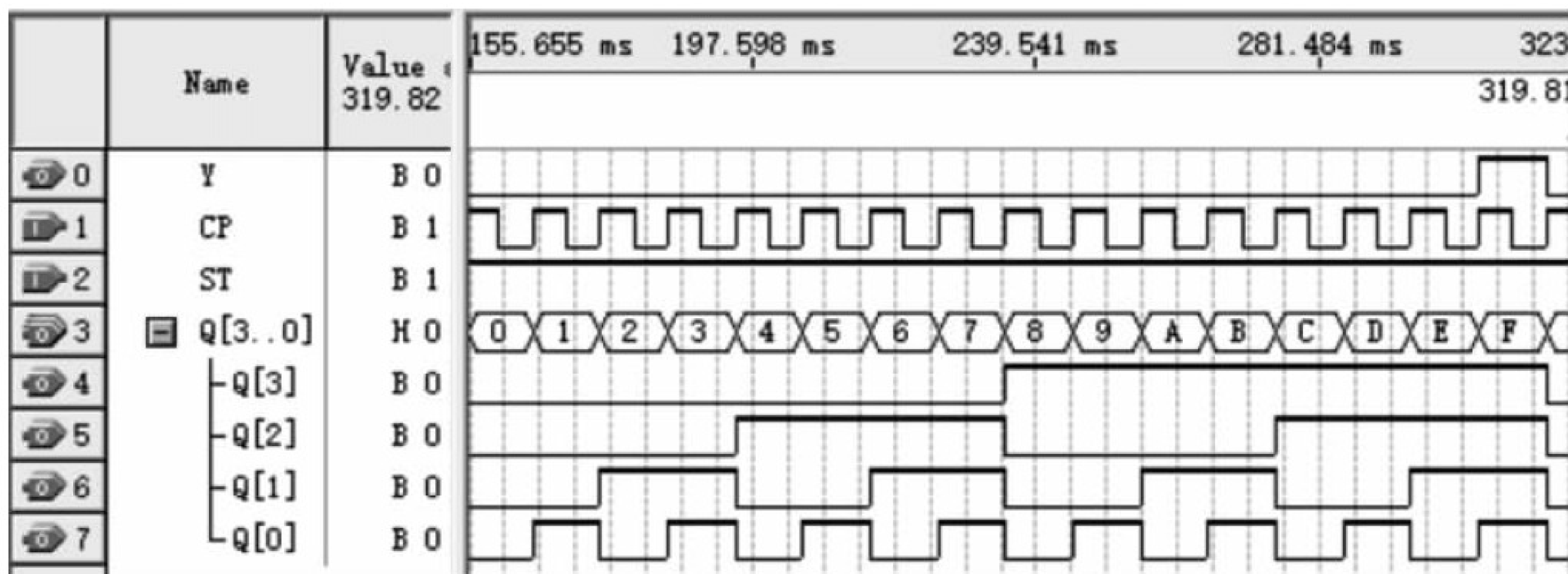


图 6.4.5　同步 4 位二进制加法计数器仿真图

从仿真图可看出，如图 6.4.4 所示电路每 16 个 CP 时钟 $Q_4Q_3Q_2Q_1$ 由状态 0 回到状态 0，为对时钟信号计数的四位二进制加法计数器，也称十六进制加法计数器。由如图 6.4.4 所示仿真时序图中输出 Y 及触发器各状态变化特点，可进一步总结该电路的逻辑特点：同步十六进制上升沿计数、满幅输出上升沿、溢出输出下降沿的加法计数器。

如图 6.4.1 所示 3 位二进制加法计数器电路驱动方程为

$$J_1 = K_1 = 1,\quad J_2 = K_2 = Q_1^n,\quad J_3 = K_3 = Q_1^nQ_2^n$$

如图 6.4.4 所示 4 位二进制加法计数器电路驱动方程为

$$J_1 = K_1 = 1,\quad J_2 = K_2 = Q_1^n,\quad J_3 = K_3 = Q_1^nQ_2^n,\quad J_4 = K_4 = Q_1^nQ_2^nQ_3^n$$

对照如图 6.4.1 和图 6.4.4 所示的 3 位、4 位二进制计数器电路，可见 4 位二进制同步加法计数器与 3 位二进制同步加法计数器的电路结构相似，4 位计数器较 3 位计数器多一级触发器，可参考如图 6.4.1 和图 6.4.4 所示电路设计更多位的二进制加法计数器。

3. 同步3位二进制减法计数器



在如图6.4.1所示电路中，下一级触发器的输入接上一级触发器的 Q 端，将其改接 $\bar{Q}$ 端，则形成另一种同步二进制计数器，具体如图6.4.6所示。

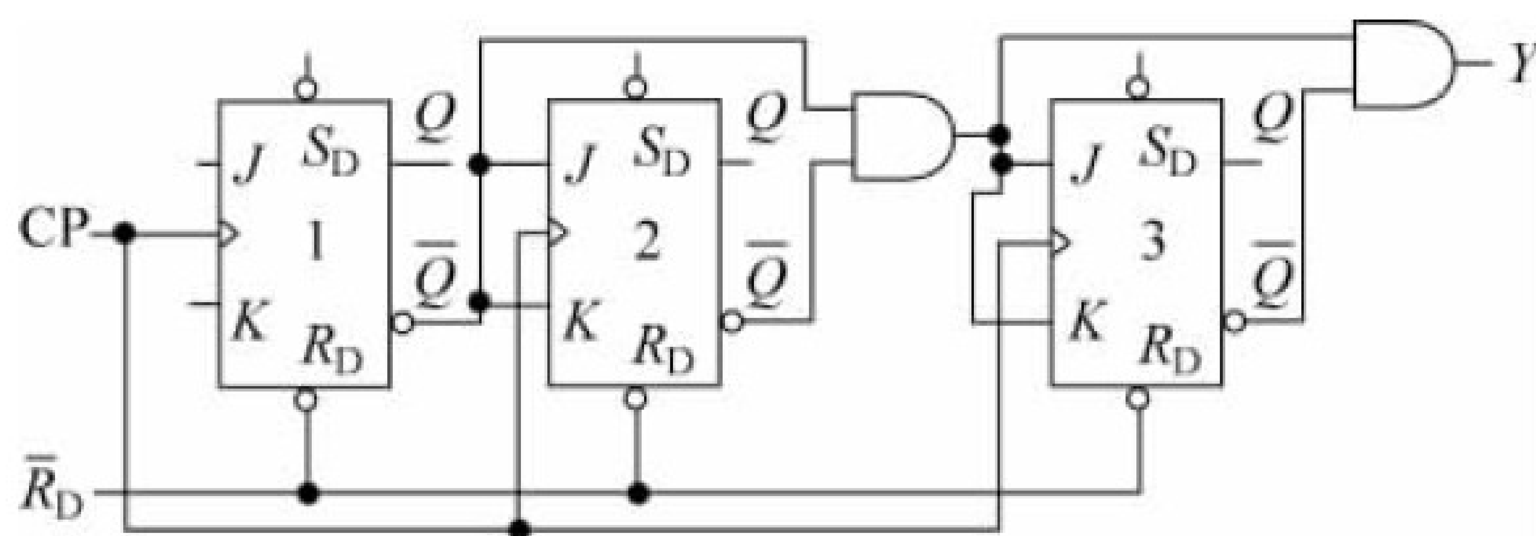


图6.4.6　3位二进制减法计数器的图

该图在Quartus Ⅱ环境中的仿真结果如图6.4.7所示。图中的数字输出形式为3个触发器按照 $Q_2Q_1Q_0$ 的顺序以总线形式仿真的结果。

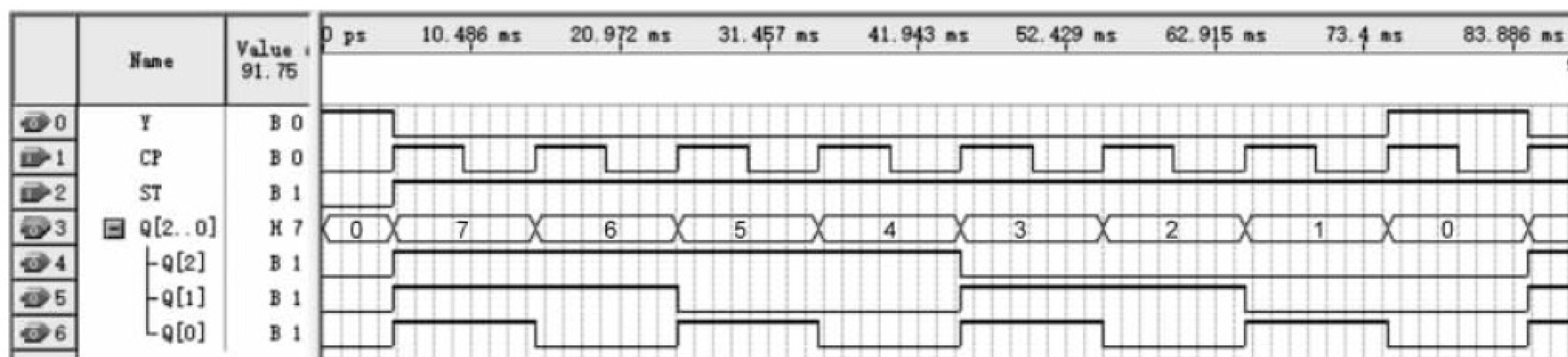


图6.4.7　如图6.4.6所示电路的仿真图

由图6.4.7可得到如图6.4.8所示的状态图。从状态图、仿真图可看出，如图6.4.6所示电路每8个CP时钟 $Q_3Q_2Q_1$ 由状态0回到状态0，为同步3位二进制减法计数器。显然，它的计数容量为8。

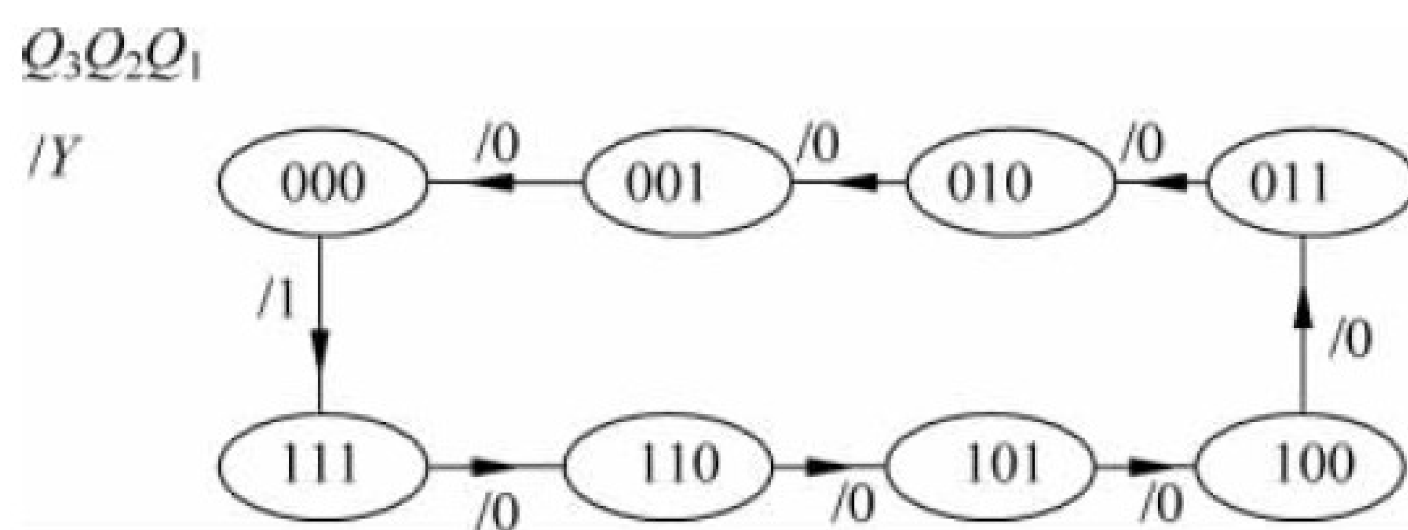


图6.4.8　如图6.4.6所示电路的状态图

类似地，也可根据JK触发器特点直接求出如图6.4.6所示电路的仿真时序图，求解思路如下。

如图6.4.6所示电路中第1个触发器连接特点：$J_1=K_1=1$，时钟端接外部时钟CP。当CP↑到来时，Q_1（仿真图中为 Q_0）翻转。可由CP波形直接画出 Q_1 的波形如图6.4.7中最下一行所示。

第2个触发器连接特点：$J_2=K_2=\bar{Q}_1^n$，时钟端接外部时钟CP。当CP↑到来且 $Q_1^n=0$

时，Q_2（仿真图中为 Q_1）翻转，其余时间保持状态不变，可由 CP 波形及 Q_1的波形直接画出 Q_2的波形，如图 6.4.7 中倒数第 2 行所示。

第 3 个触发器连接特点：$J_3=K_3=\bar{Q}_1^n\bar{Q}_2^n$，时钟端接外部时钟 CP。当 CP↑到来且 $Q_1^n=Q_2^n=0$ 时，Q_3（仿真图中为 Q_2）翻转，其余时间保持状态不变，可由 CP 波形及 Q_1、Q_2的波形直接画出 Q_3的波形，如图 6.4.7 中倒数第 3 行所示。

由如图 6.4.7 所示仿真时序图输出 Y 及触发器各状态变化特点，可进一步总结如图 6.4.6 所示电路的逻辑特点：同步上升沿计数、零幅输出上升沿、溢出输出下降沿的八进制减法计数器。

如图 6.4.6 所示 3 位二进制减法计数器电路驱动方程为

$$J_1=K_1=1,\quad J_2=K_2=\bar{Q}_1^n,\quad J_3=K_3=\bar{Q}_1^n\bar{Q}_2^n$$

结合上面的驱动方程，可参照更多位的二进制加法计数器电路构成方法构成更多位的二进制减法计数器电路。

4. 同步 3 位二进制可逆计数器

对照如图 6.4.1 和图 6.4.6 所示 3 位二进制电路，从第 2 个触发器开始，二者的接法差别：加法计数器 J、K 接 Q，减法计数器 J、K 接 $\bar{Q}$。

基于上面的接法特点，从第 2 个触发器开始，将同步二进制加法计数器和同步二进制减法计数器合并在一起，由控制信号 M 加以控制。

当 $M=1$ 时，按加法进行计数，J、K 接 Q，写成表达式，有

$$J=K=MQ \tag{6.4.1}$$

当 $M=0$ 时，按减法进行计数，J、K 接 $\bar{Q}$，有

$$J=K=\bar{M}\bar{Q} \tag{6.4.2}$$

综上所述，同步可逆计数器第 2 个触发器的驱动方程为

$$J=K=MQ+\bar{M}\bar{Q}=\overline{\overline{MQ}\;\overline{\bar{M}\bar{Q}}} \tag{6.4.3}$$

电路如图 6.4.9 所示。

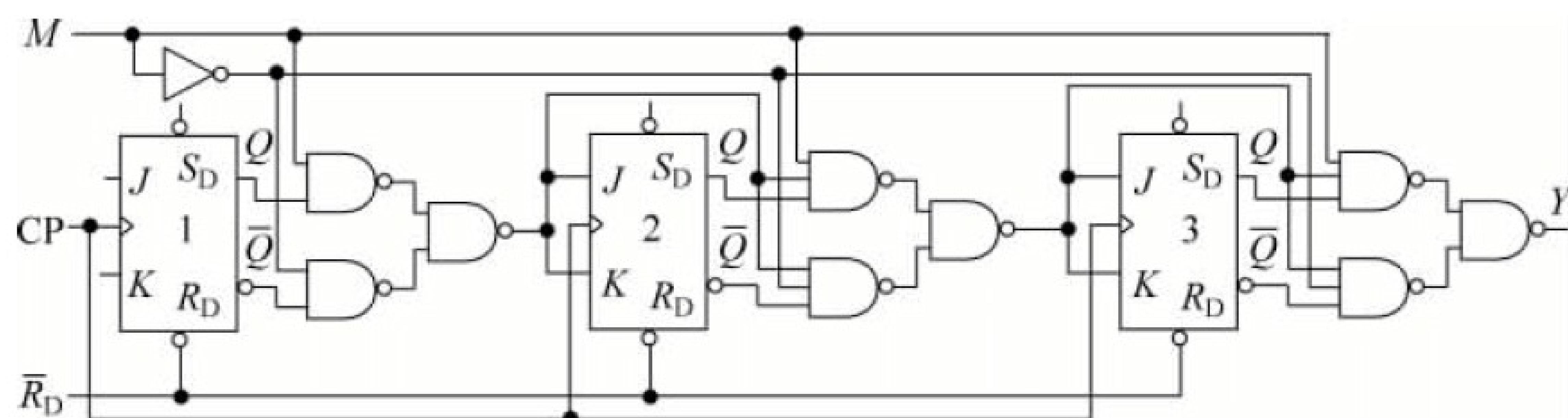


图 6.4.9 同步可逆 3 位二进制计数器

可参照更多位的二进制加法计数器电路构成方法构成更多位的二进制可逆计数器电路。

5. 同步十进制加法计数器

同步十进制计数器电路实例如图 6.4.10 所示。该图在 Quartus Ⅱ环境中的仿真结果如图 6.4.11 所示。图中的数字输出形式为 4 个触发器按照 $Q_3Q_2Q_1Q_0$ 的顺序以总线形式仿真的结果。

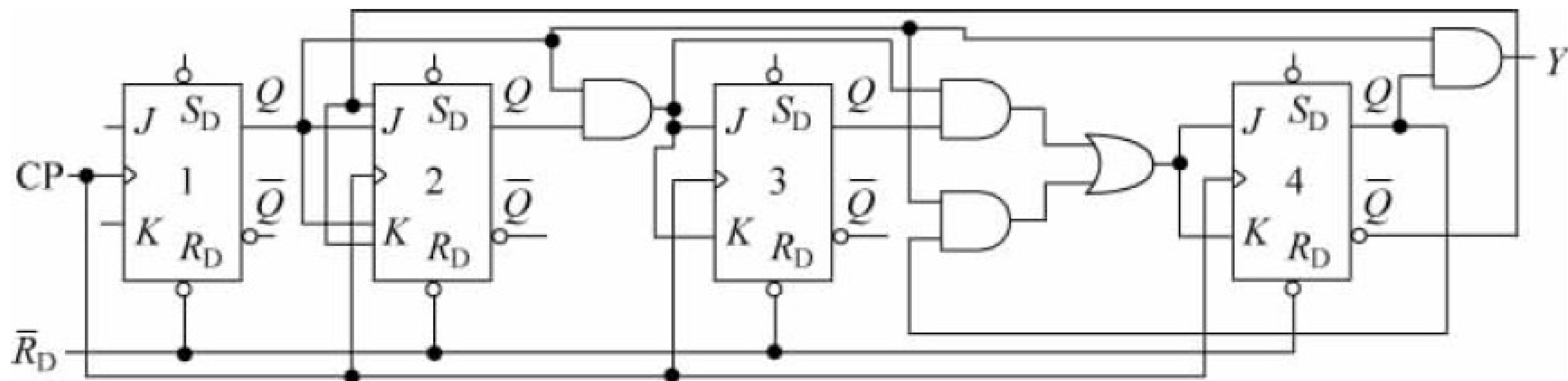


图 6.4.10　同步十进制计数器的图

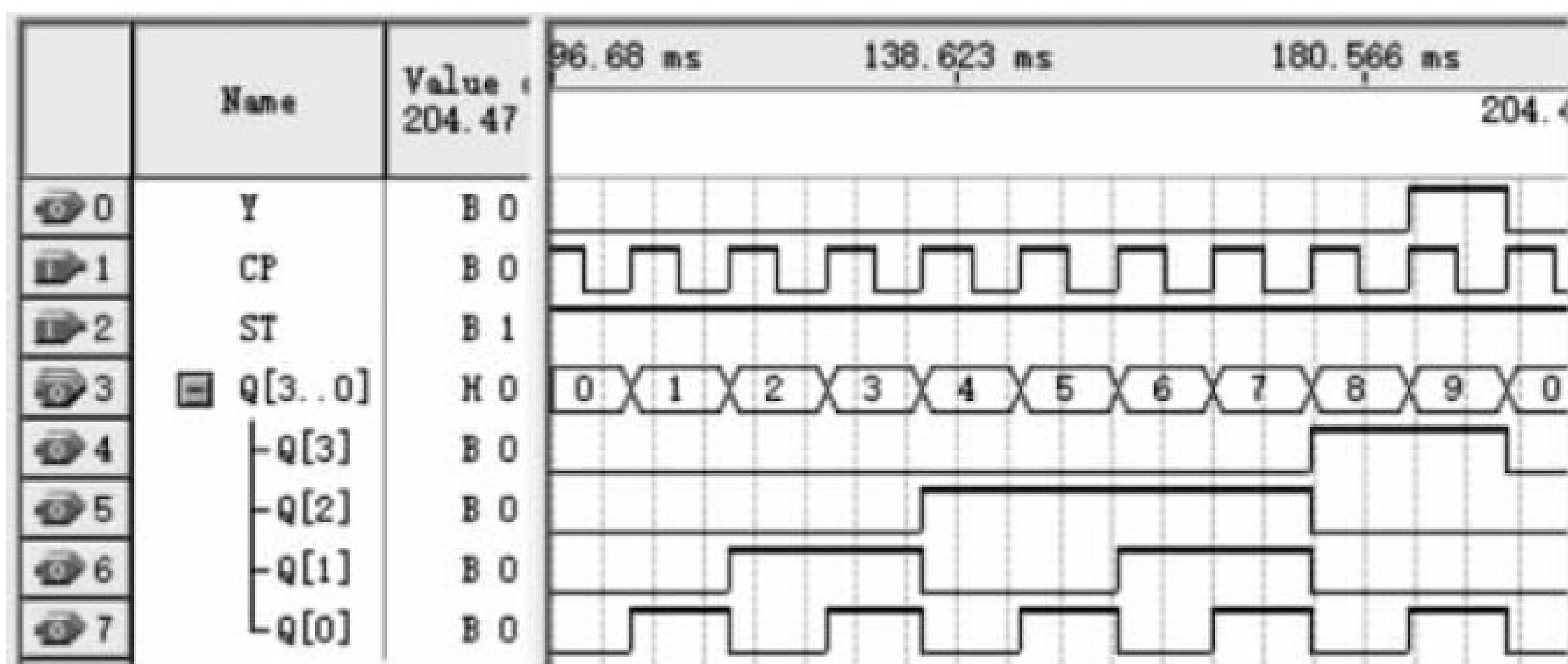


图 6.4.11　如图 6.4.10 所示电路的仿真图

由图 6.4.11 可得到如图 6.4.12 所示的状态图。从状态图、仿真图可看出，如图 6.4.10 所示电路每 10 个 CP 时钟 $Q_4Q_3Q_2Q_1$ 由状态 0 回到状态 0，为对时钟信号计数的十进制加法计数器。由如图 6.4.11 所示仿真时序图中输出 Y 及触发器各状态变化特点，可进一步总结该电路的逻辑特点：同步十进制上升沿计数、满幅输出上升沿、溢出输出下降沿的加法计数器。

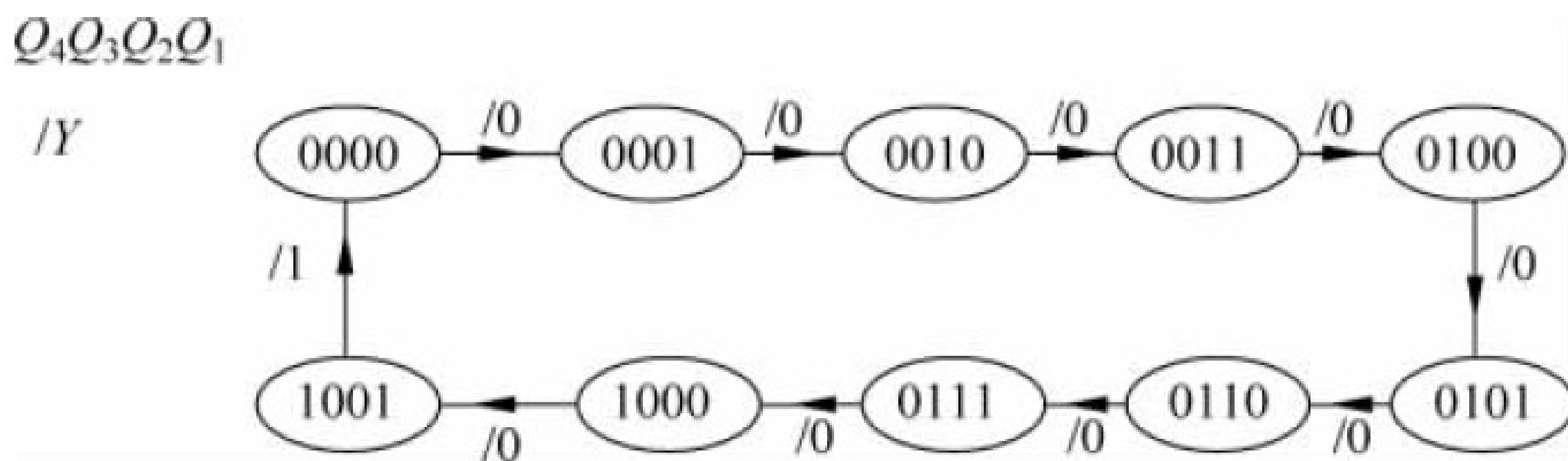


图 6.4.12　如图 6.4.10 所示电路的状态图

本书主要采用计算机仿真分析各时序电路，也可采用 6.2 节介绍的时序逻辑电路分析方法分析时序电路。

下面以如图 6.4.10 所示同步十进制计数器为例介绍利用时序逻辑电路分析方法分析计数器电路的方法。

（1）写出电路的驱动方程、输出方程。驱动方程为

$$\begin{cases} J_1 = K_1 = 1 \\ J_2 = K_2 = Q_1^n \bar{Q}_4^n \\ J_3 = K_3 = Q_1^n Q_2^n \\ J_4 = K_4 = Q_1^n Q_2^n Q_3^n + Q_1^n Q_4^n \end{cases} \tag{6.4.4}$$

输出方程为

$$Y = Q_1^n Q_4^n \tag{6.4.5}$$

（2）写出电路的状态方程。将驱动方程代入 JK 触发器的特性方程 $Q^{n+1} = J\bar{Q}^n + \bar{K}Q^n$ 中，就得到了电路的状态方程，即

$$\begin{cases} Q_1^{n+1} = \bar{Q}_1^n \\ Q_2^{n+1} = Q_1^n \bar{Q}_4^n \bar{Q}_2^n + \overline{Q_1^n \bar{Q}_4^n} Q_2^n \\ Q_3^{n+1} = Q_1^n Q_2^n \bar{Q}_3^n + \overline{Q_1^n Q_2^n} Q_3^n \\ Q_4^{n+1} = (Q_1^n Q_2^n Q_3^n + Q_1^n Q_4^n) \bar{Q}_4^n + \overline{Q_1^n Q_2^n Q_3^n + Q_1^n Q_4^n} Q_4^n \end{cases} \tag{6.4.6}$$

（3）画出电路的状态图。由状态方程可得到如图 6.4.12 所示的状态图。

（4）检查电路能否自启动。4 个触发器有 16 个状态，尚有 6 个无效状态，应检查电路能否自启动。

设电路的原状态分别为“1010”“1100”“1110”，当 CP 脉冲到来时，将电路的初始状态代入状态方程和输出方程，有

$$\begin{cases} 1010 \xrightarrow{/0} 1011 \xrightarrow{/1} 0110 \\ 1100 \xrightarrow{/0} 1101 \xrightarrow{/1} 0100 \\ 1110 \xrightarrow{/0} 1111 \xrightarrow{/1} 0010 \end{cases}$$

（5）结论。可见，如图 6.4.10 所示电路为同步能够自启动的十进制加法计数器。

通过上面的分析不难发现，十进制和二进制计数器在原理上并无本质区别，逻辑特点也大体相同。类似地，同步十进制计数器也有减法、可逆等多种类型，有兴趣的读者可参考相关书籍。

当然，大多数工程实践中的计数器既不是二进制计数器，也不是十进制计数器，而是任意的 N 进制的计数器。

N 进制计数器一般用集成二进制或十进制计数器结合门电路构成，具体将在后面的内容中介绍。

6.4.3　常用中规模集成同步计数器

中规模集成同步计数器的产品型号比较多。其电路结构是在基本计数器如二进制计数器、十进制计数器的基础上增加一些附加电路，以扩展其功能。

常用的集成二进制同步计数器有加法计数器和可逆计数器两种类型。

1. 集成 4 位二进制同步加法计数器

集成 4 位二进制同步加法计数器的主要产品有 CT54161/CT74161、CT54LS161/CT74LS161、CC40161 等，它们采用的都是异步清零（又称为异步清除）。此外还有用同步清零的计数器，它们是当 $\overline{CR}$ 低电平有效时，在时钟信号作用下实现清零，如 CT54163/CT74163、CT54LS163/CT74LS163、CC40163 等 4 位二进制计数器。

下面介绍比较典型的芯片 CT54LS161/CT74LS161①（4 位二进制计数器）的控制逻辑及其应用。

（1）引脚说明。如图 6.4.13 所示为集成 4 位二进制计数器 74LS161。图中，CP 为输入的计数脉冲，也就是加到各个触发器的时钟信号端的时钟脉冲。$\overline{CR}$ 是清零端，$\overline{LD}$ 是置数控制端；CT_P 和 CT_T 是两个计数器工作状态的控制端，$D_0 \sim D_3$ 是并行输入数据端，CO 是进位信号输出端，$Q_0 \sim Q_3$ 是计数器状态输出端。

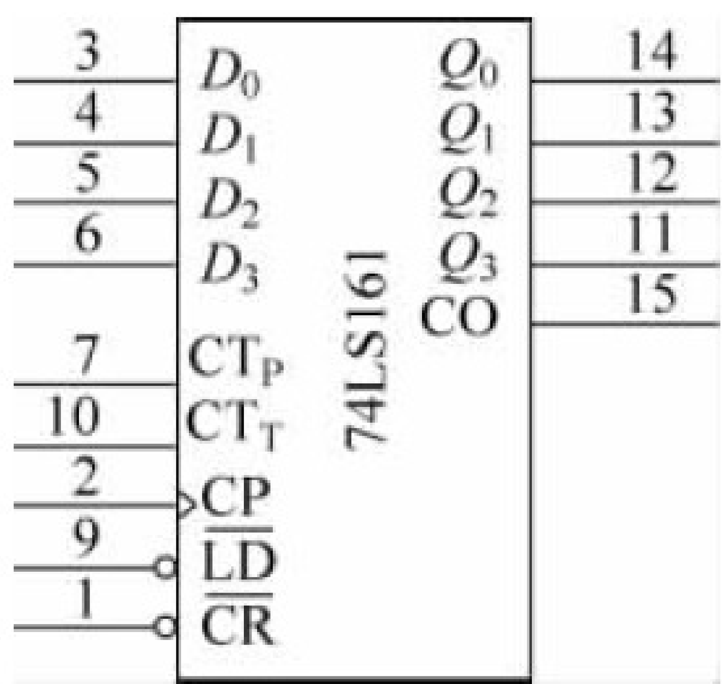


图 6.4.13　74LS161

（2）功能表。如表 6.4.1 所示为 74LS161 集成 4 位二进制计数器的功能表。由表 6.4.1 所示功能表可以清楚地看出，集成 4 位二进制同步加法计数器具有如下功能：

① 异步清零功能。当 $\overline{CR}=0$ 时，计数器清零。从表中第 1 行可以看出，$\overline{CR}=0$，其他输入信号都不起作用，由时钟触发器的逻辑特性知道，其异步输入端的信号是优先的，$\overline{CR}=0$ 正是通过 $\overline{R}_D=0$ 使各个触发器清零的。这一工作又称为**计数器的复位**。

表 6.4.1　74LS161 功能表

$\overline{CR}$	$\overline{LD}$	CT_P	CT_T	CP	$D_0\ D_1\ D_2\ D_3$	$Q_0\ Q_1\ Q_2\ Q_3$
0	×	×	×	×	× × × ×	0 0 0 0
1	0	×	×	↑	$d_0\ d_1\ d_2\ d_3$	$d_0\ d_1\ d_2\ d_3$
1	1	1	1	↑	× × × ×	正常计数
1	1	×	0	×	× × × ×	保持（但 CO=0）
1	1	0	1	×	× × × ×	保持

② 同步并行置数功能。从表中第 2 行可以看出，当 $\overline{CR}=1$，$\overline{LD}=0$ 时，在 CP 脉冲上升沿的操作下，并行输入数据端 $D_0 \sim D_3$ 输入的数据 $d_0 \sim d_3$ 置入计数器，使计数器的状态为 $Q_3Q_2Q_1Q_0=d_3d_2d_1d_0$。

③ 二进制同步加法计数功能。从表中第 3 行可以看出，当 $\overline{CR}=\overline{LD}=1$ 时，若 $CT_T=CT_P=1$，则 4 位二进制加法计数器对输入的 CP 计数脉冲进行加法计数。当第 15 个 CP 脉

① CT54/CT74 系列芯片与 CT54LS/CT74LS 系列对应芯片的逻辑功能、工作原理、引线排列图、逻辑符号都相同。在后面的内容中，不再区分 CT54/CT74 系列与 CT54LS/CT74LS 系列芯片，统一采用 74LS TTL 系列的命名方法。

冲到来时，计数器的状态 $Q_3Q_2Q_1Q_0$ 为“1111”，同时进位信号 CO=1；当第 16 个 CP 脉冲到来时，计数器的状态 $Q_3Q_2Q_1Q_0$ 为“0000”，同时进位信号 CO 跳变到 0，计数器向高一位产生下降沿输出信号。

④ 保持功能。从表中第 4、5 行可以看出，当$\overline{CR}=\overline{LD}=1$ 时，若 $CT_T \cdot CT_P=0$，则计数器将保持原来状态不变。

对于进位输出 CO 有两种情况：如果 $CT_T=0$，那么 CO=0；如果 $CT_T=1$，则 CO 保持原始状态不变。

综上所述，表 6.4.1 所示的功能表反映了 74LS161 是一个具有**异步清零、同步置数、可以保持状态不变的 4 位二进制同步上升沿加法计数器**。

当然，集成芯片的逻辑功能是通过电路来实现的。74LS161 的电路原理图如图 6.4.14 所示。

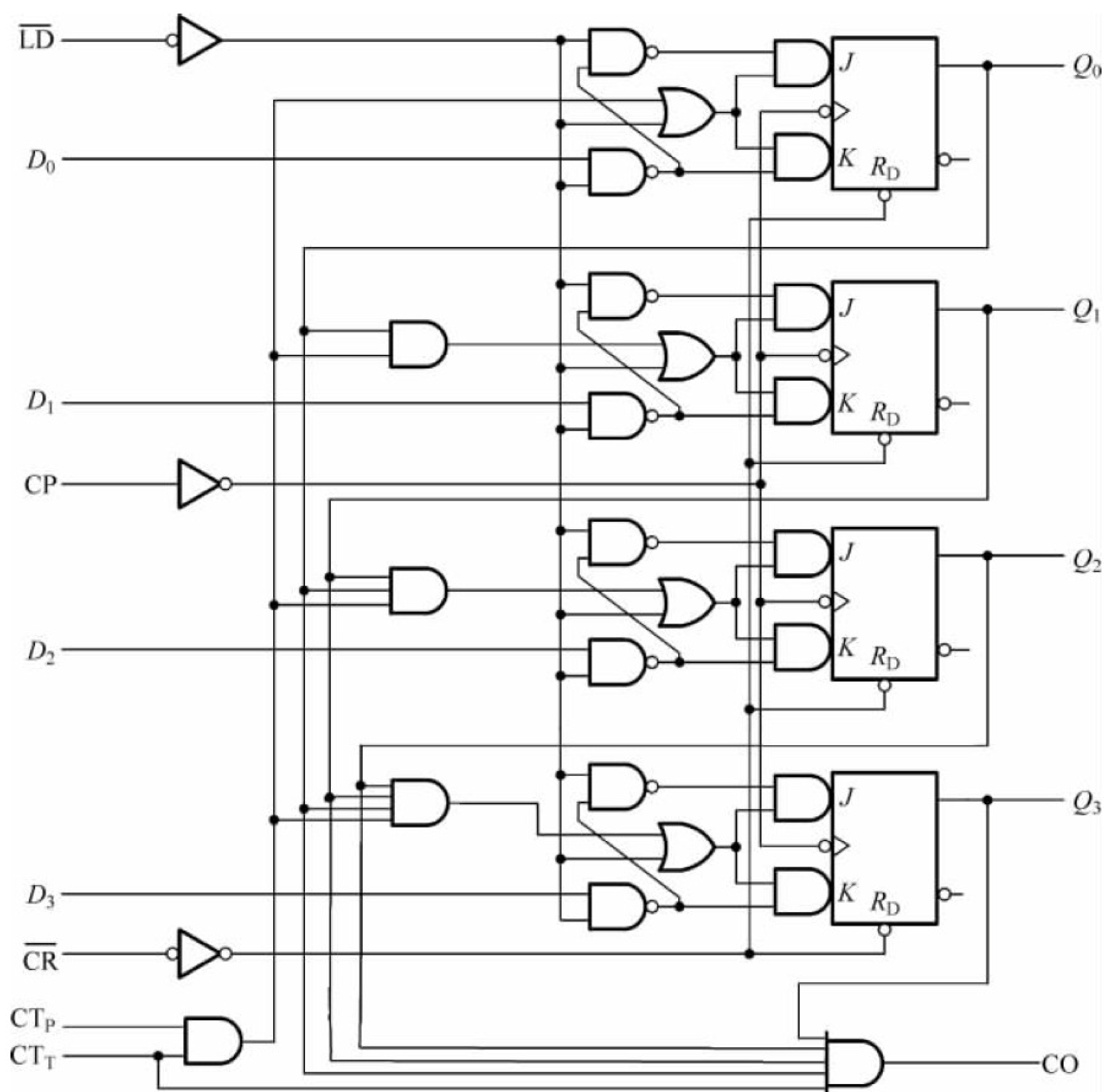


图 6.4.14　74LS161 的电路原理图

图 6.4.14 所示电路使用下降沿触发的 JK 触发器，外部时钟经反相器取反后接触发器的时钟输入端。图中，当$\overline{CR}$、$\overline{LD}$无效，CT_P和 CT_T有效时，有

$$J_0=K_0=1,\quad J_1=K_1=Q_0^n,\quad J_2=K_2=Q_1^nQ_0^n,\quad J_3=K_3=Q_2^nQ_1^nQ_0^n$$

此时，如图 6.4.14 所示电路具有和如图 6.4.4 所示电路相同的驱动方程。可见，如图 6.4.14 所示的 74LS161 的电路原理图是在如图 6.4.4 所示的 4 位二进制加法计数器电路原理图的基础上，增加了一些辅助功能（如置数功能、保持功能等）而构成的同步二进制计数器。

根据加法计数器电路特点，74LS161 的进位 CO 也具有满幅输出上升沿、溢出输出下降沿的应用特点。

在如图 6.4.14 所示的 74LS161 的电路原理图中，初始数据 $D_3D_2D_1D_0$ 通过置数输入控制端 $\overline{LD}$ 加到 JK 触发器的 J、K 输入端，只有在时钟信号上升沿到来时才会改变触发器的状态。这种在 CP 脉冲上升沿的作用下将外部数据置入计数器的置数方式称为同步置数。有兴趣的读者可进一步分析 74LS161 的电路原理图，进一步理解异步清零、同步置数的动作特点及其电路实现原理。

74LS161 的清零方式为异步清零。当清零信号 $\overline{CR}=0$ 时，74LS161 各触发器立即进入状态“0”。也有支持同步清零的集成 4 位二进制同步加法计数器，参考芯片如 74LS163，其功能表如表 6.4.2 所示。

表 6.4.2　74LS163 功能表

$\overline{CR}$	$\overline{LD}$	CT_P	CT_T	CP	$D_0\ D_1\ D_2\ D_3$	$Q_0\ Q_1\ Q_2\ Q_3$
0	×	×	×	↑	× × × ×	0 0 0 0
1	0	×	×	↑	$d_0\ d_1\ d_2\ d_3$	$d_0\ d_1\ d_2\ d_3$
1	1	1	1	↑	× × × ×	正常计数
1	1	×	0	×	× × × ×	保持（CO＝0）
1	1	0	1	×	× × × ×	保持

从表中第 1 行可以看出，在 $\overline{CR}=0$ 的情况下，当 CP 脉冲上升沿到来时，计数器的状态 $Q_3Q_2Q_1Q_0=0000$。这种在 CP 脉冲上升沿的作用下将计数器置“0”的清零方式称为同步清零。

74LS163 除了采用同步清零方式外，其逻辑功能、计数工作原理和引线排列图与 74LS161 没有区别。

常用的 CMOS 集成同步计数器有 CC4520、CC4526（减法计数器）等，有兴趣的读者可参阅有关书籍。

2. 集成十进制同步加法计数器

集成十进制同步加法器种类较多，TTL 产品有 74LS160、74LS162 等，CMOS 产品有 CC40160 等。

74LS160 是一个具有异步清零、同步置数、可以保持状态不变的十进制同步加法计数器。74LS162 与 74LS160 的区别是 74LS162 采用同步清零方式，即当 $\overline{CR}=0$ 时，只有 CP 脉冲上升沿到来，计数器才被清零。74LS160、74LS161、74LS162、74LS163 的输出端排列图和逻辑符号完全相同，其逻辑功能也基本类似，其区别如表 6.4.3 所示。读者可对照

74LS161、74LS163 的逻辑功能理解 74LS160、74LS162。

表 6.4.3 74LS160、74LS161、74LS162、74LS163 加法计数器功能简表

74LS161	74LS160	74LS163	74LS162
异步清零	异步清零	同步清零	同步清零
同步置数	同步置数	同步置数	同步置数
状态保持	状态保持	状态保持	状态保持
十六进制计数	十进制计数	十六进制计数	十进制计数

74LS160 的电路原理图如图 6.4.15 所示。该电路使用下降沿触发的 JK 触发器，外部时钟经反相器取反后接触发器的时钟输入端。图中，当$\overline{CR}$、$\overline{LD}$无效，CT_P和 CT_T有效时，有

$$J_1 = K_1 = 1,\quad J_2 = K_2 = Q_1^n\overline{Q}_4^n$$

$$J_3 = K_3 = Q_1^nQ_2^n,\quad J_4 = K_4 = Q_1^nQ_2^nQ_3^n + Q_1^nQ_4^n$$

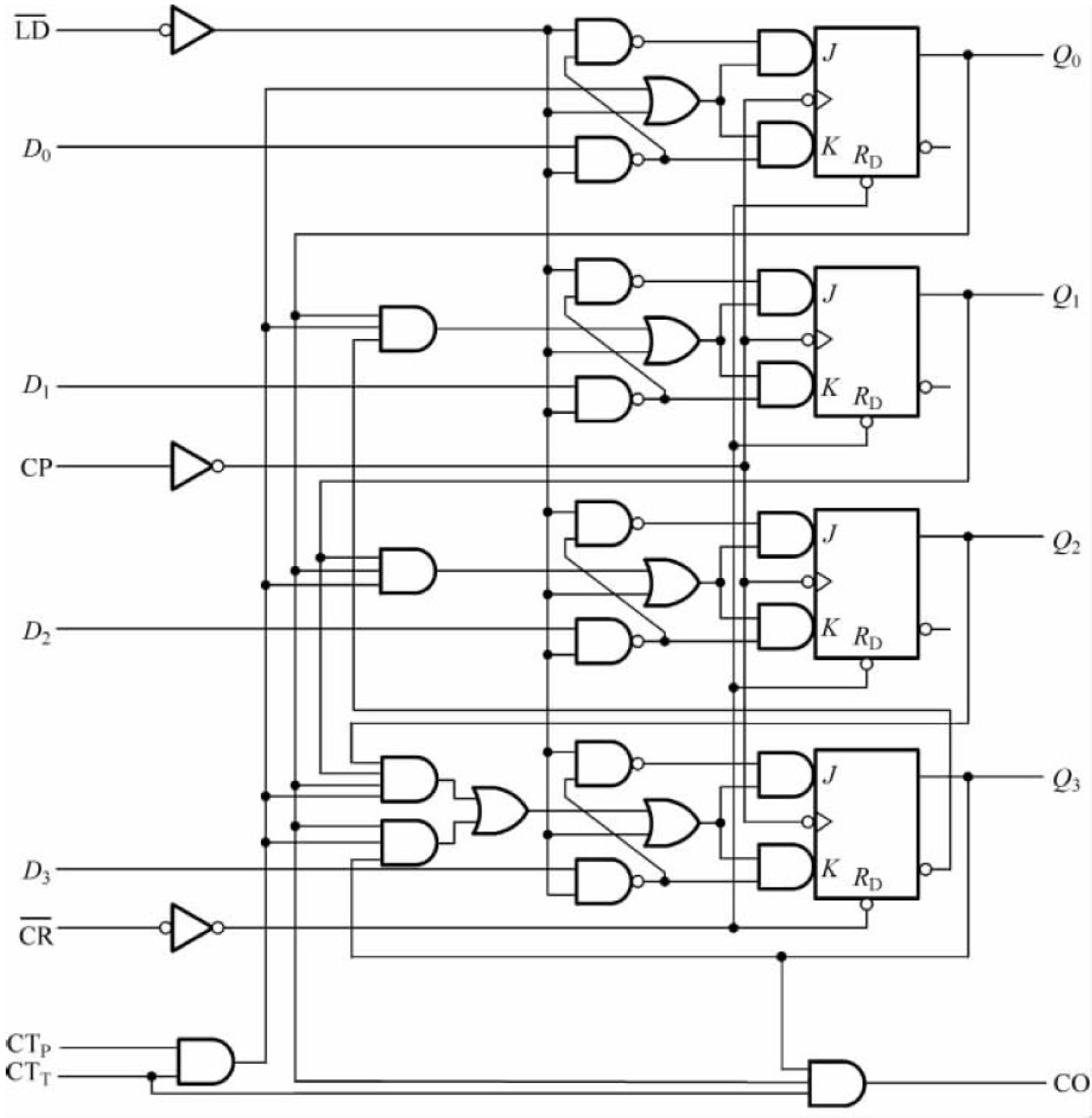


图 6.4.15 74LS160 的电路原理图

此时，如图 6.4.15 所示电路具有和如图 6.4.10 所示电路相同的驱动方程。可见，如图 6.4.15 所示的 74LS160 的原理电路是在如图 6.4.10 所示的十进制加法计数器原理电

路的基础上，增加一些辅助功能(如置数功能、保持功能等)而构成的同步二进制计数器。

根据加法计数器电路特点，74LS160 的进位 CO 也具有满幅(值为 9)输出上升沿、溢出输出下降沿的应用特点。

74LS160 的清零方式为异步清零。74LS162 则支持同步清零。74LS162 除了采用同步清零方式外，其逻辑功能、计数工作原理和引线排列图和 74LS160 没有区别。

3. 集成同步可逆计数器



集成 4 位二进制同步可逆计数器，有单时钟和双时钟两种类型。

(1) 单时钟的集成 4 位二进制同步可逆计数器。下面以比较典型的单时钟集成 4 位二进制同步可逆计数器 74LS191 为例，做简单的说明。

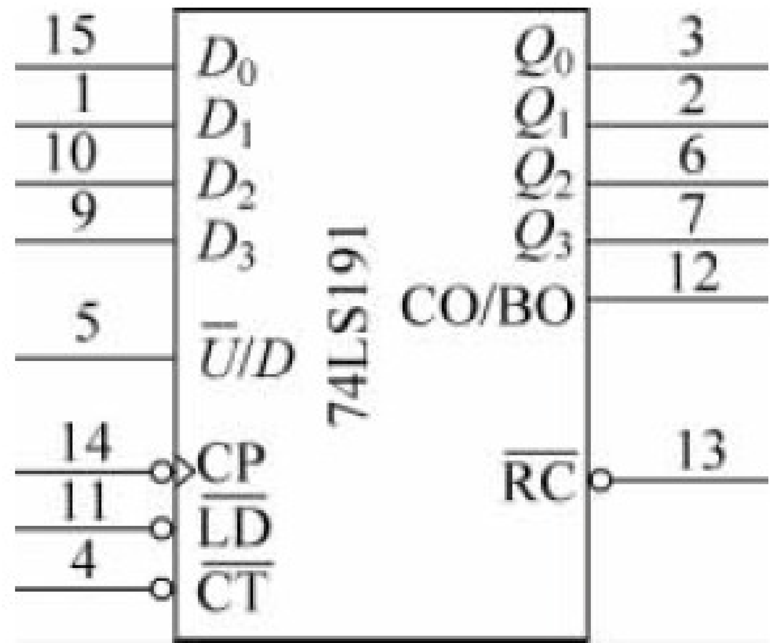


图 6.4.16　74LS191

① 引脚说明。如图 6.4.16 所示为集成 4 位二进制同步可逆计数器 74LS191。图中，$\overline{U}/D$ 为加减计数控制端，$\overline{\mathrm{CT}}$为使能端，$\overline{\mathrm{LD}}$为异步置数端，$D_0 \sim D_3$ 为并行数据输入端，$Q_0 \sim Q_3$ 为计数器计数状态输出端，CO/BO 为进位/借位信号输出端，$\overline{\mathrm{RC}}$为级联端，用于多个芯片级联。

根据加法计数器电路特点，74LS191 的进位 CO 具有满幅输出上升沿、溢出输出下降沿的应用特点。当 74LS191 进行减法计数时，CO 引脚为借位输出 BO，具有零幅输出上升沿、溢出输出下降沿的应用特点，因此，当 74LS191 加法计数到最大值全 1 或减法计数到最小值零时，CO/BO 输出高电平。

② 功能表。如表 6.4.4 所示为 74LS191 集成 4 位二进制同步可逆计数器的功能表。该表反映出的功能为：同步可逆计数功能(表中第 2、3 行)；异步并行置数功能(表中第 1 行)；保持功能(表中第 4 行)。74LS191 集成 4 位二进制同步可逆计数没有专用的清零输入端，但是可以借助 $D_0 \sim D_3$端异步并行置入数据 0000，间接实现清零功能。

表 6.4.4　74LS191 功能表

$\overline{\mathrm{LD}}$	$\overline{\mathrm{CT}}$	$\overline{U}/D$	CP	$D_0\ D_1\ D_2\ D_3$	$Q_0\ Q_1\ Q_2\ Q_3$
0	×	×	×	$d_0\ d_1\ d_2\ d_3$	$d_0\ d_1\ d_2\ d_3$
1	0	0	↑	× × × ×	加法计数
1	0	1	↑	× × × ×	减法计数
1	1	×	×	× × × ×	保持

74LS191 的电路原理图如图 6.4.17 所示。图中，初始数据 $D_3D_2D_1D_0$是通过置数输入控制端$\overline{\mathrm{LD}}$直接加到 JK 触发器的异步置数输入端的。当置数信号$\overline{\mathrm{LD}}=0$ 时，74LS191 各触发器立即进入初始预设数据状态，这种置数方式称为异步置数。

也可通过该电路原理图进一步理解 74LS191 $\overline{\mathrm{RC}}$端的特点。

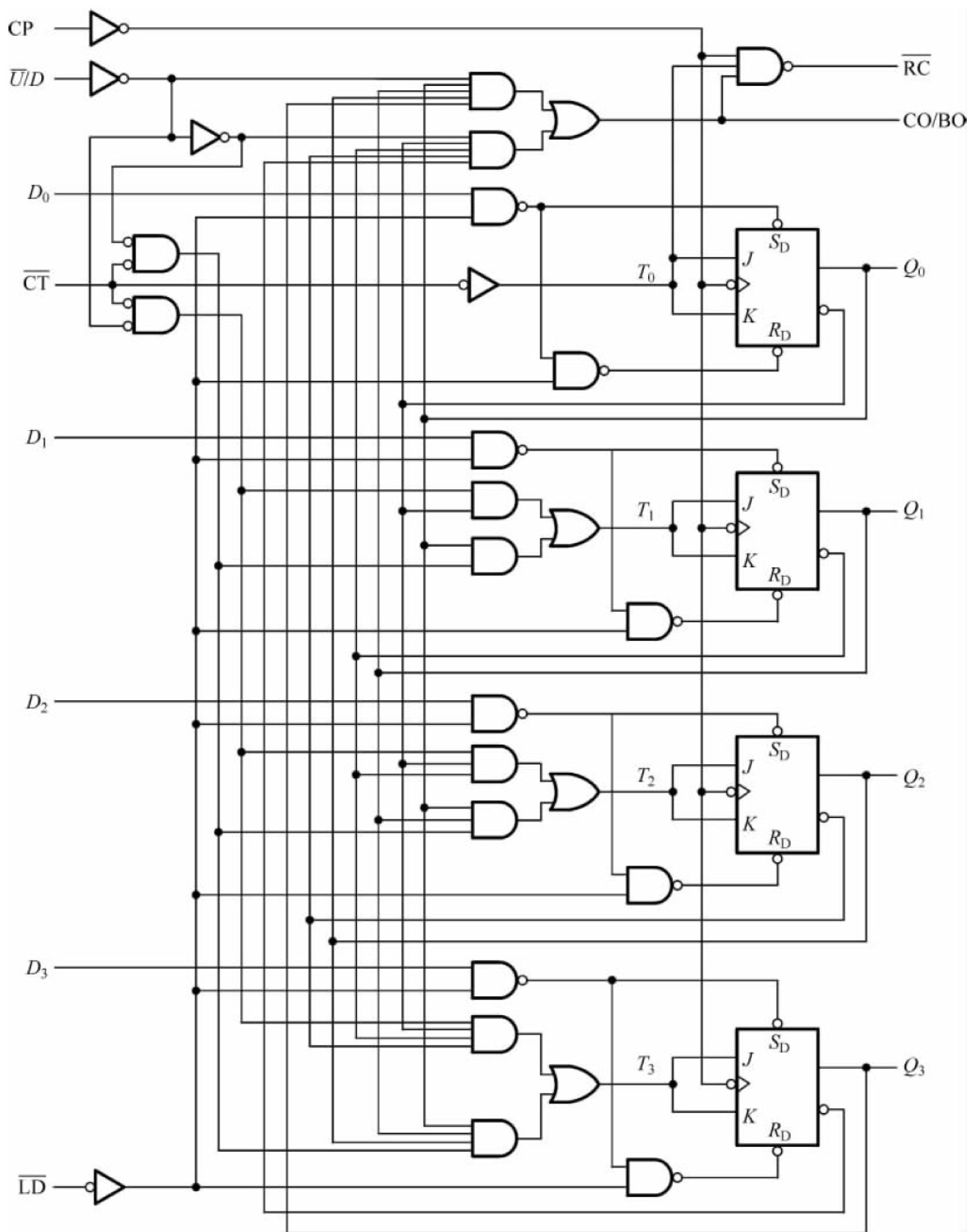


图 6.4.17　74LS191 的电路原理图

$\overline{RC}$端主要在多片可逆计数器级联时使用，由如图 6.4.17 所示的 74LS191 原理电路可写出$\overline{RC}$端表达式为

$$\overline{RC} = \overline{\overline{CP} \cdot CO/BO \cdot CT}$$

当$\overline{CT}$＝0(即 CT＝1)，CO/BO＝1 时，$\overline{RC}$＝CP，$\overline{RC}$端产生的输出进位脉冲的波形和输入计数脉冲的波形相同。即当计数允许，进位或借位信号有效时，输出计数时钟信号。

可通过下面的例题来进一步理解。

【例 6.4.1】 电路如图 6.4.18(a)所示，请分析如图 6.4.18(b)所示输入下相应的输出波形。

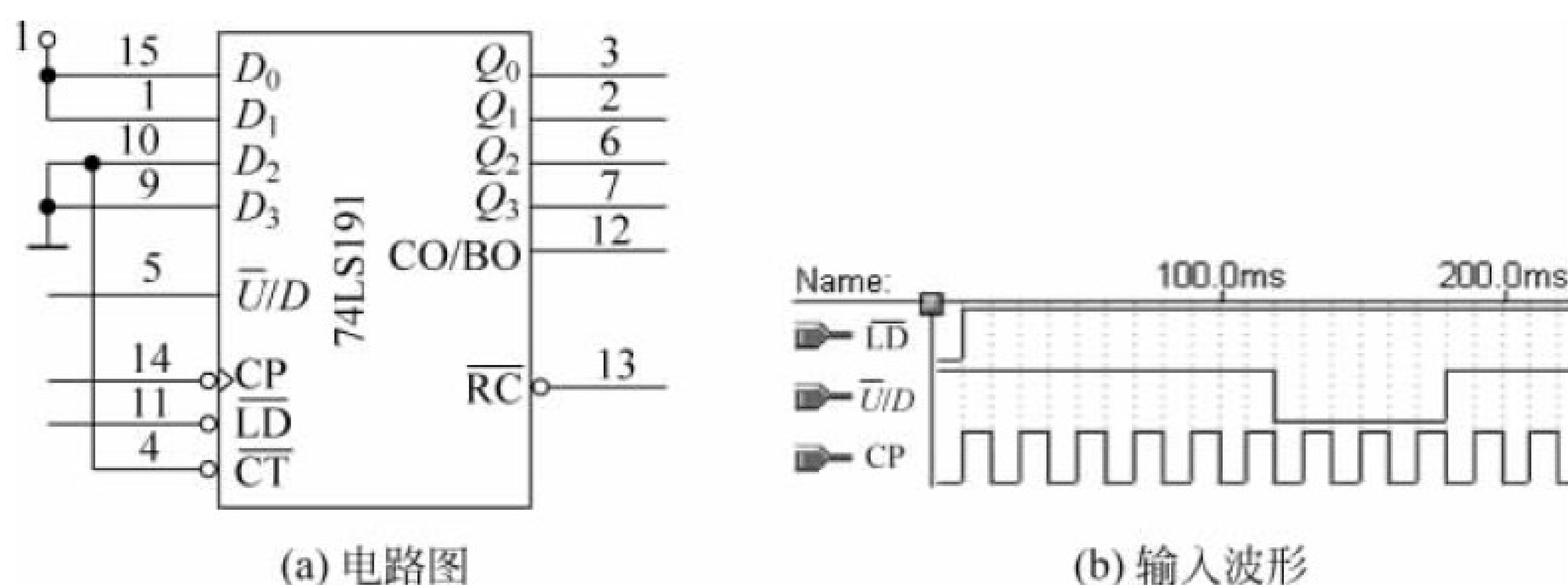


图 6.4.18 例 6.4.1 的图

解 仿真图如图 6.4.19 所示，图中，CLK 对应时钟信号 CP，ST 对应$\overline{LD}$，out 对应 CO/BO，RC 对应$\overline{RC}$，UD 对应 $\overline{U}/D$，简要分析如下：

电路初始数据 $D_3D_2D_1D_0=0011$，异步置数 ST 有效，计数器进入状态“3”；UD=1，先进行减法计数；从“3”开始经 5 个脉冲减法计数到“E”。之后 UD=0，进行加法计数；从“E”开始经 3 个脉冲加法计数到“1”。之后，UD=1，进行减法计数，可求得计数器状态如图 6.4.19 最下一行所示。

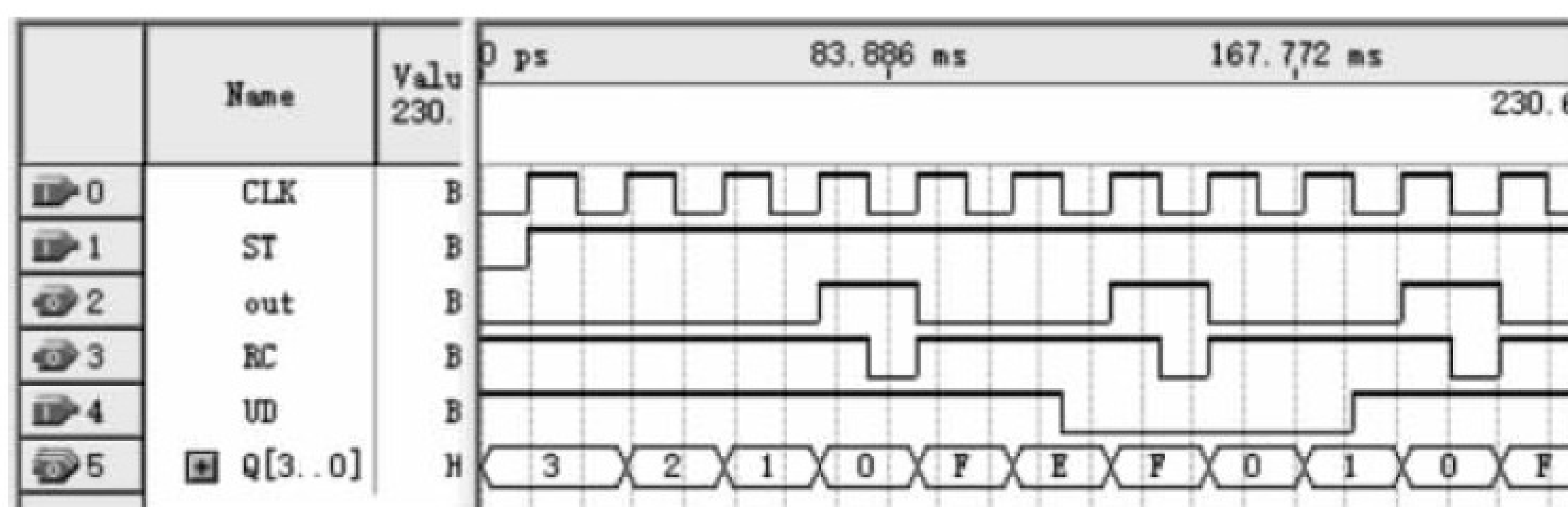


图 6.4.19 例 6.4.1 的仿真图

根据加法、减法计数电路的特点，74LS191 加法计数到“F”时，进位信号 CO=1；减法计数到“0”时，借位信号 BO=1。可根据计数器状态的变化规律求出 CO/BO 的波形，如图 6.4.19 第 3 行所示。

图 6.4.19 中，计数允许一直有效，当进位或借位信号有效时，$\overline{RC}$端输出计数时钟信号。可求出$\overline{RC}$端波形，如图 6.4.19 第 4 行所示。

由图 6.4.19 可知，当计数器溢出时，进位/借位信号输出下降沿，不适合直接作为计数脉冲信号。但计数器溢出时，$\overline{RC}$端输出上升沿，因此，当多片 74LS191 集成计数器级联时，只需将低位的$\overline{RC}$端与高位的 CP 端连接起来，各片芯片的 $\overline{U}/D$、$\overline{CT}$、$\overline{LD}$端连接在一起就可以了。

其他的集成单时钟 4 位二进制同步可逆计数器还有 74LS169、CC4516 等，有兴趣的读者可参考相关书籍。

(2) 双时钟的集成 4 位二进制同步可逆计数器。双时钟的集成 4 位二进制同步可逆计数器具有加法、减法两个计数时钟，常用于加法计数脉冲和减法计数脉冲来自两个不同脉

冲源的计数应用场合。下面以比较典型的双时钟集成4位二进制同步可逆计数器74LS193为例做简单说明。

① 引脚说明。如图6.4.20所示为双时钟集成4位二进制同步可逆计数器74LS193。图中，CR是异步清零端，高电平有效；$\overline{LD}$为异步置数控制端；CP_U为加法计数脉冲输入端，CP_D为减法计数脉冲输入端；$\overline{CO}$为进位脉冲输出端，$\overline{BO}$为借位脉冲输出端；$D_0 \sim D_3$为并行数据输入端，$Q_0 \sim Q_3$为计数器的状态输出端。

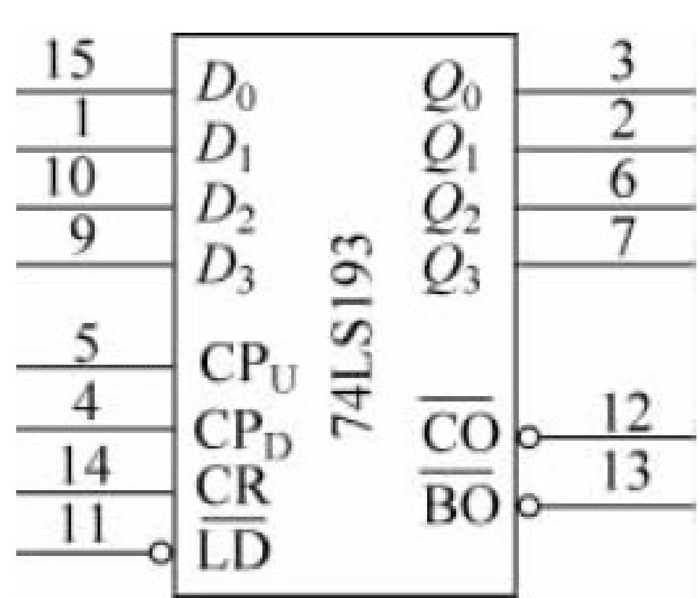


图6.4.20　74LS193

② 功能表。如表6.4.5所示为74LS193的功能表。该表所反映的功能有异步清零功能(表中第1行)、异步并行置数功能(表中第2行)、同步可逆计数功能(表中第3、4行反映：当CP_U计数脉冲到来时按加法计数，当CP_D计数脉冲到来时按减法计数)、保持功能(表中第5行)。

表6.4.5　74LS193功能表

CR	$\overline{LD}$	CP_U	CP_D	$D_0\ D_1\ D_2\ D_3$	$Q_0\ Q_1\ Q_2\ Q_3$
1	×	×	×	× × × ×	0 0 0 0
0	0	×	×	$d_0\ d_1\ d_2\ d_3$	$d_0\ d_1\ d_2\ d_3$
0	1	↑	1	× × × ×	加法计数
0	1	1	↑	× × × ×	减法计数
0	1	1	1	× × × ×	保持

从表6.4.5可以看出，异步清零CR优先级高于异步并行置数$\overline{LD}$。还可以看出，74LS193没有专门的计数允许端，加法计数脉冲CP_U兼做减法计数的计数允许，$CP_U=1$时，允许减法计数；减法计数脉冲CP_D兼做加法计数的计数允许，$CP_D=1$时，允许加法计数。

③ $\overline{CO}$、$\overline{BO}$。$\overline{CO}$、$\overline{BO}$分别为进位、借位脉冲输出端，同时可供多片双时钟可逆计数器级联时使用。

可通过下面的例题来进一步理解。

【例6.4.2】 已知74LS193的预置数$D_3D_2D_1D_0=0011$，另外4个输入端(ST对应$\overline{LD}$，RD对应CR)的输入波形如图6.4.21所示，请分析计数器的状态变化规律及进位、借位信号输出波形。

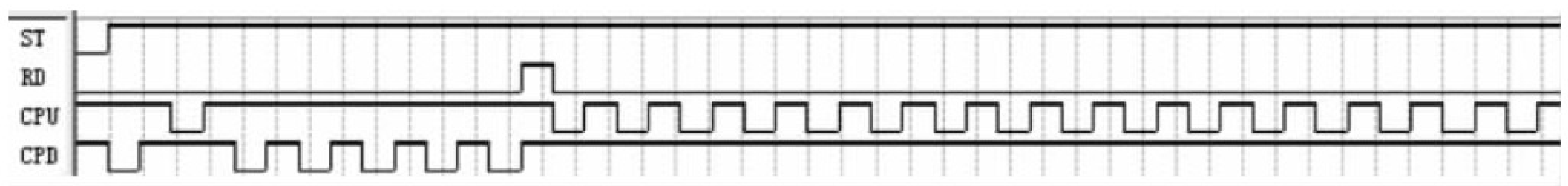


图6.4.21　例6.4.2的输入波形

解　仿真图如图6.4.22所示，图中，ST对应$\overline{LD}$，RD对应CR，简要分析如下。

电路初始数据$D_3D_2D_1D_0=0011$，异步置数ST有效，计数器进入状态"3"；之后有1个减法计数上升沿，加法计数脉冲为1，减法计数允许，计数器进入状态"2"；接着有1个加法

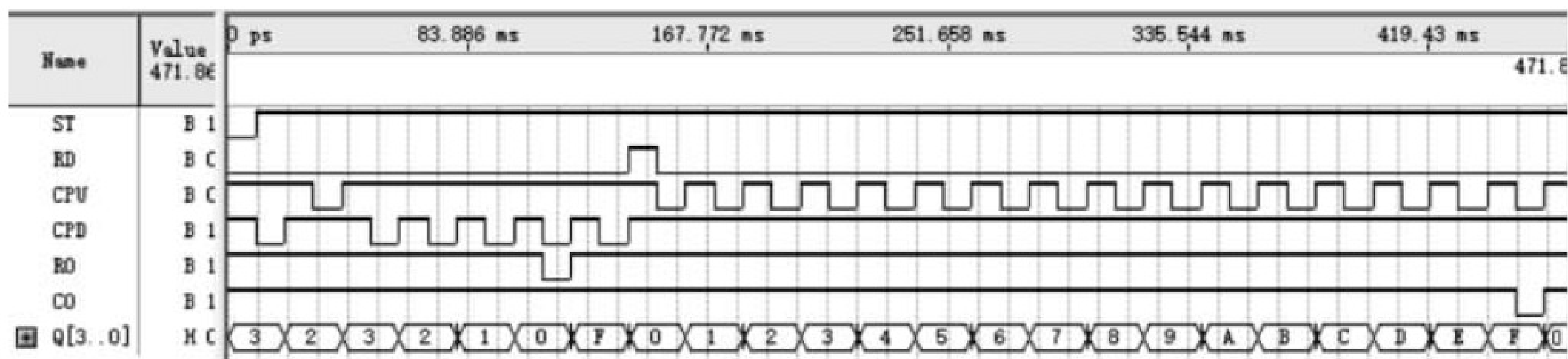


图 6.4.22 例 6.4.2 的仿真波形

计数上升沿，计数器进入状态“3”；从“3”开始经 4 个减法脉冲计数到“F”；之后，异步清零信号 CR=1，计数器立即进入状态“0”；之后，从“0”开始经 16 个加法脉冲计数器回到“0”，可求得计数器状态如图 6.4.22 最下一行所示。

根据加法、减法计数特点，74LS193 加法计数到“F”时进位信号有效，减法计数到“0”时借位信号有效。从仿真波形可看出，进位（借位）信号有效时为低电平，计数器溢出时，进位/借位信号输出上升沿，具有和时钟信号相同的波形，因此，当多片 74LS193 集成计数器级联时，只需将低位的$\overline{CO}$端、$\overline{BO}$端分别与高位的 CP_U 端、CP_D 端连接起来，各片芯片的 CR 端连接在一起，$\overline{LD}$端连接在一起。

（3）集成十进制同步可逆计数器。与集成二进制同步可逆计数器一样，集成十进制同步可逆计数器也有单时钟和双时钟两种类型。常用的产品型号有 74LS190、74LS192、74LS168、CC4510 等。

集成十进制同步可逆计数器 74LS190 与集成十六进制同步可逆计数器 74LS191 的输出端排列图和逻辑符号相同，其区别是前者为十进制计数器，后者为十六进制计数器。双时钟集成十进制同步可逆计数器 74LS192 的输出端排列和符号与 74LS193 集成双时钟 4 位二进制同步可逆计数器的输出端排列图和逻辑符号相同，其区别如表 6.4.6 所示。读者可对照 74LS191、74LS193 的逻辑功能理解 74LS190、74LS192。

表 6.4.6 74LS190、74LS191、74LS192、74LS193 可逆计数器功能简表

74LS191	74LS190	74LS193	74LS192
异步置数	异步置数	异步置数、清零	异步置数、清零
单时钟	单时钟	双时钟	双时钟
状态保持	状态保持	状态保持	状态保持
进位/借位、级联输出	进位/借位、级联输出	进位、借位输出	进位、借位输出
十六进制计数	十进制计数	十六进制计数	十进制计数

6.4.4 异步计数器的电路构成特点

异步计数器不同于同步计数器。组成异步计数器的各级触发器时钟脉冲，不全是计数输入脉冲，所以各级触发器的状态转移不是在同一时钟脉冲作用下同时产生转移。

1. 异步3位二进制减法计数器



构成异步二进制计数器方法较多，这里介绍一种通用方法。异步3位二进制减法计数器实例如图6.4.23所示，图中各触发器均为上升沿触发。该图在Quartus Ⅱ环境中的仿真结果如图6.4.24所示。图中的数字输出形式为3个触发器按照$Q_2Q_1Q_0$（电路中为$Q_3Q_2Q_1$）的顺序以总线形式仿真的结果。

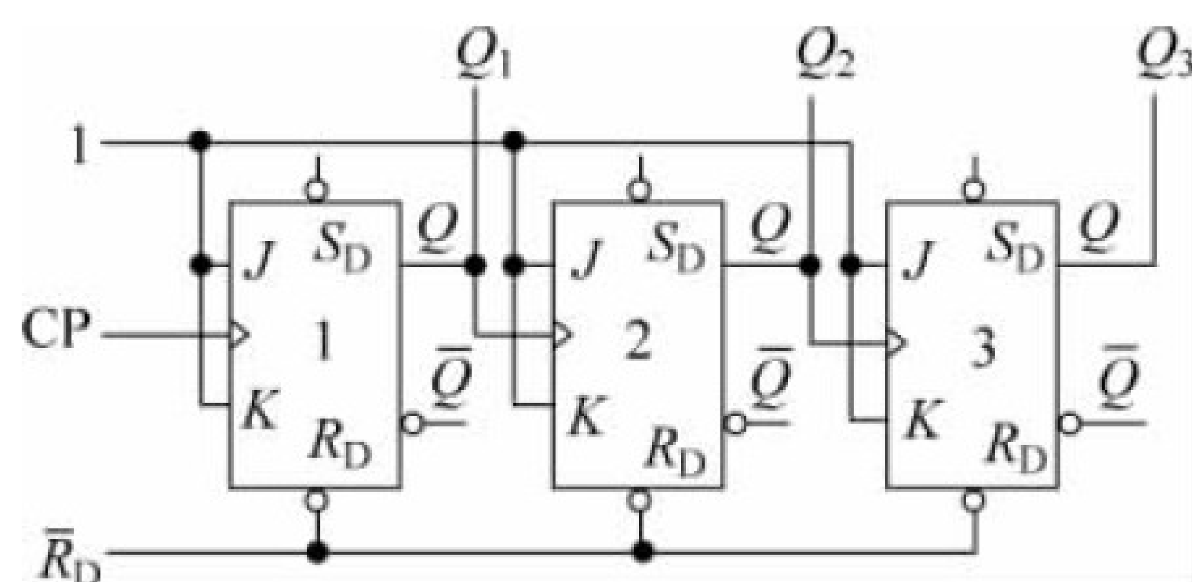


图6.4.23　异步3位二进制减法计数器

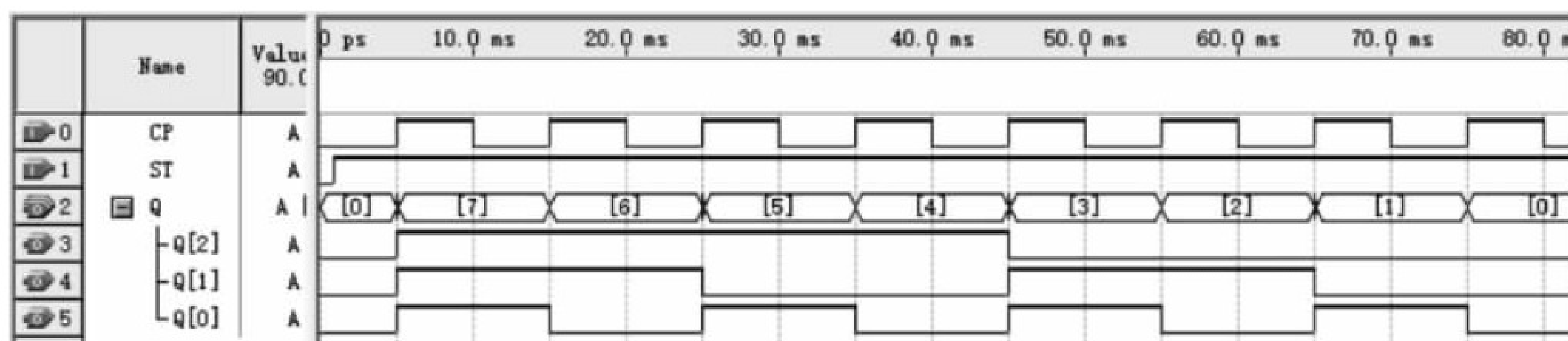


图6.4.24　如图6.4.23所示电路的仿真图

由图6.4.24可得到如图6.4.25所示的状态图。从状态图、仿真图可看出，如图6.4.23所示电路每8个CP时钟$Q_3Q_2Q_1$由状态0回到状态0，为对时钟信号计数的3位二进制减法计数器，也称八进制减法计数器。

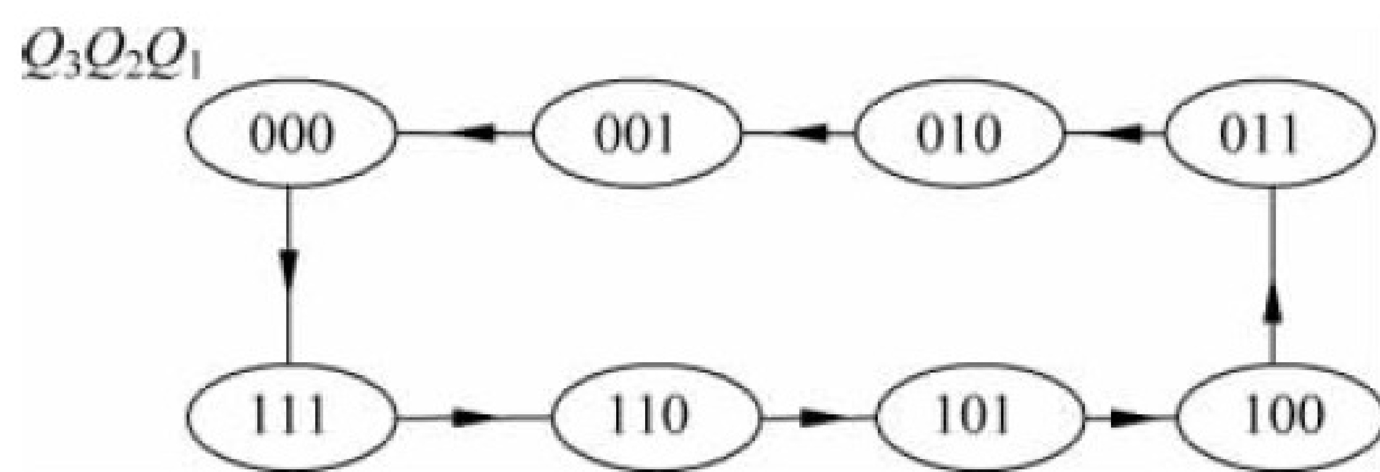


图6.4.25　如图6.4.23所示电路的状态图

也可根据JK触发器特点直接求出如图6.4.23所示电路的仿真时序图，求解思路如下。

如图6.4.23所示电路中第1个触发器连接特点：$J_1=K_1=1$，时钟端接外部时钟CP。当CP↑到来时，Q_1（仿真图中对应Q_0）翻转。可由CP波形直接画出Q_1的波形如图6.4.24中最下一行所示。

第2个触发器连接特点：$J_2=K_2=1$，时钟端接Q_1。当Q_1↑到来时，Q_2（仿真图中对应Q_1）翻转，其余时间保持状态不变，可由Q_1的波形直接画出Q_2的波形如图6.4.24中倒数第2行所示。

第 3 个触发器连接特点：$J_3=K_3=1$，时钟端接 Q_2。当 Q_2↑到来时，Q_3（仿真图中对应 Q_2）翻转，其余时间保持状态不变，可由 Q_2 的波形直接画出 Q_3 的波形，如图 6.4.24 中倒数第 3 行所示。

从上面的状态图、仿真图上看，如图 6.4.23 所示的异步 3 位二进制减法计数器和如图 6.4.6 所示的同步 3 位二进制减法计数器从逻辑功能到动作特点似乎都没有区别。

进一步分析如图 6.4.24 所示仿真图，该图的时钟脉冲周期为 10ms（Quartus Ⅱ环境中的“GRID SIZE”为 5ms）。若将 Quartus Ⅱ环境中的“GRID SIZE”改为 50ns，则其仿真结果如图 6.4.26 所示，其局部（全 0 到全 1 的变化过程）放大如图 6.4.27 所示。

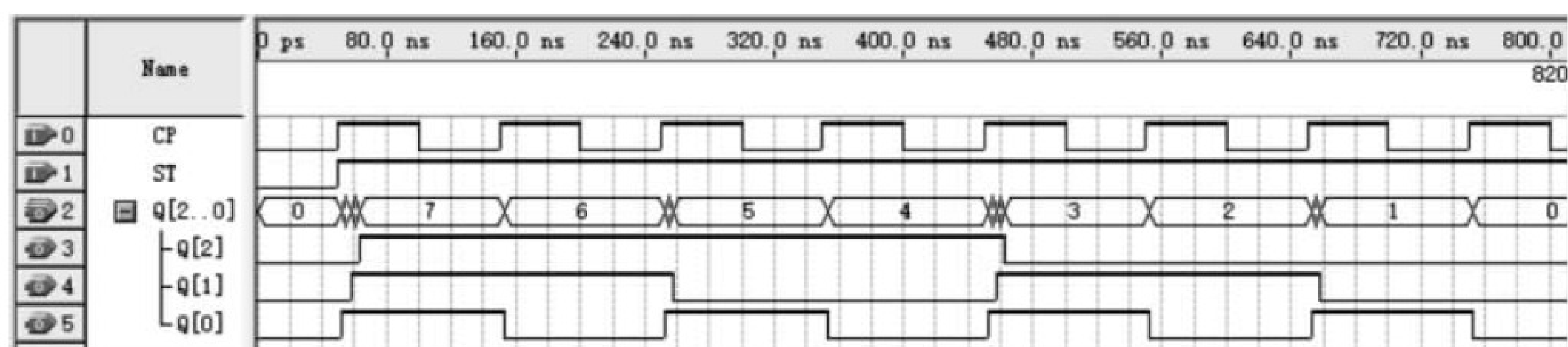


图 6.4.26 GRID SIZE 为 50ns 的仿真结果图

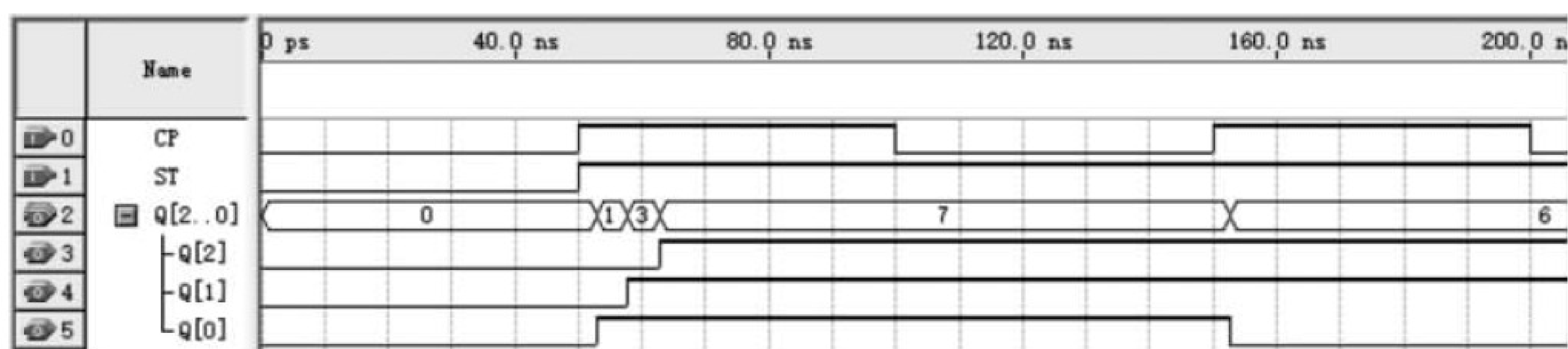


图 6.4.27 全 0 到全 1 的变化过程图

通过图 6.4.27 所示可看出，如图 6.4.23 所示的异步 3 位二进制减法计数器由状态“0”进入到状态“7”时，中间经过了“1”“3”两个过渡状态。

为什么这样呢？

分析如图 6.4.23 所示电路，只有第 1 个触发器的时钟输入与外部时钟相连，当外部时钟上升沿到来时，只有第 1 个触发器由状态“0”翻转到“1”，计数器进入过渡状态“1”；与此同时，第 1 个触发器 Q 端将产生上升沿，第 2 个触发器将由状态“0”翻转到“1”，计数器进入过渡状态“3”；与此同时，第 2 个触发器 Q 端将产生上升沿，之后，第 3 个触发器由状态“0”翻转到“1”，计数器进入稳定状态“7”。

显然，异步电路有中间状态，响应速度慢，可靠性差。

2. 异步 3 位二进制加法计数器



当然，异步二进制计数器也存在加法、可逆计数器电路。在图 6.4.23 中，若将触发器 1、2 的 $\overline{Q}$ 分别接触发器 2、3 的时钟（图 6.4.28，各触发器均为上

升沿计数器)，则构成异步的 3 位二进制加法计数器，该图在 Quartus Ⅱ环境中的仿真结果如图 6.4.29 所示。图中的数字输出形式为 3 个触发器按照 $Q_2Q_1Q_0$（电路中为 $Q_3Q_2Q_1$）的顺序以总线形式仿真的结果。

从仿真图可看出，如图 6.4.28 所示电路每 8 个 CP 时钟 $Q_3Q_2Q_1$ 由状态 0 回到状态 0，为对时钟信号计数的 3 位二进制加法计数器，也称八进制加法计数器。

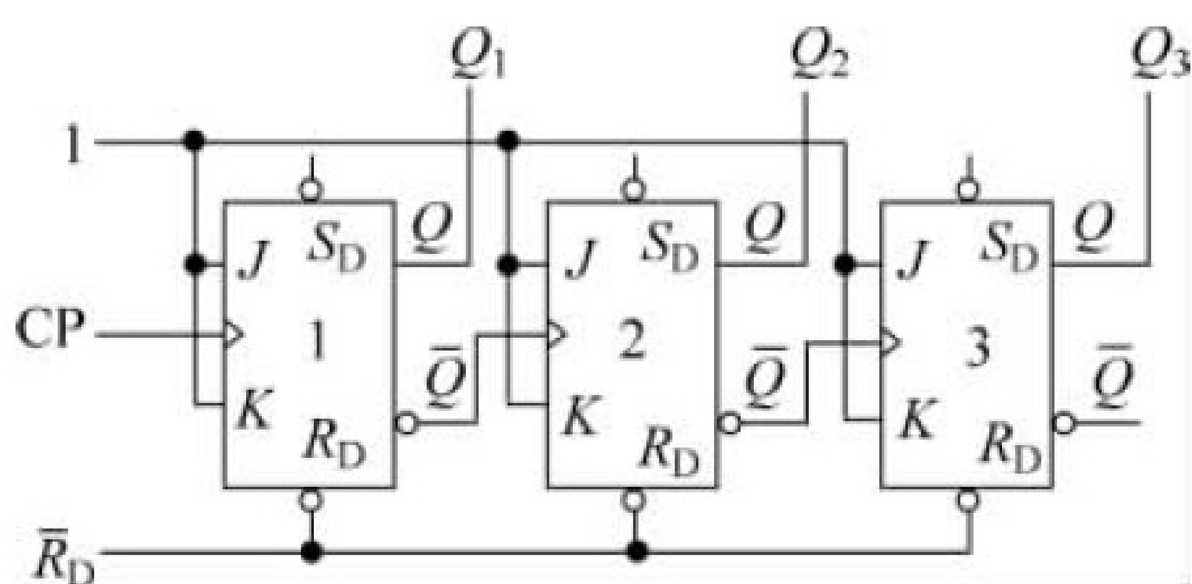


图 6.4.28　加法计数器电路

也可根据 JK 触发器特点直接求出如图 6.4.29 所示电路的仿真时序图，求解思路如下。

如图 6.4.28 所示电路中第 1 个触发器连接特点：$J_1=K_1=1$，时钟端接外部时钟 CP。当CP↑到来时，Q_1（仿真图中对应 Q_0）翻转。可由 CP 波形直接画出 Q_1 的波形如图 6.4.29 中最下一行所示。

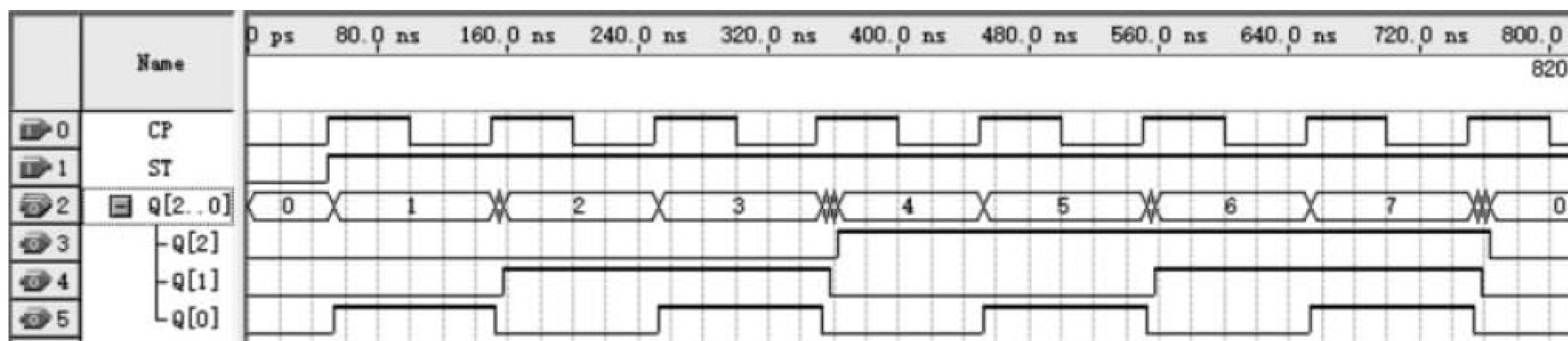


图 6.4.29　如图 6.4.28 所示电路的仿真图

第 2 个触发器连接特点：$J_2=K_2=1$，时钟端接 $\bar{Q}_1$。当 Q_1↓（$\bar{Q}_1$↑）到来时，Q_2（仿真图中对应 Q_1）翻转，其余时间保持状态不变，可由 Q_1 的波形直接画出 Q_2 的波形，如图 6.4.29 中倒数第 2 行所示。

第 3 个触发器连接特点：$J_3=K_3=1$，时钟端接 $\bar{Q}_2$。当 Q_2↓（$\bar{Q}_2$↑）到来时，Q_3（仿真图中对应 Q_2）翻转，其余时间保持状态不变，可由 Q_2 的波形直接画出 Q_3 的波形，如图 6.4.29 中倒数第 3 行所示。



3. 异步计数器电路特点总结

对照如图 6.4.23 和图 6.4.28 所示异步减法、加法计数器电路，可总结异步计数器电路构成特点：对上升沿触发的 JK 触发器，从第 2 个触发器开始，时钟端接 Q 为减法计数器，时钟端接 $\bar{Q}$ 为加法计数器。可根据如图 6.4.23 和图 6.4.28 所示异步减法、加法计数器电路的结构特点构成更多位的加法、减法计数器。将加法、减法计数器合在一起，可构成可逆计数器。

此外，异步计数器的逻辑功能与触发器的动作特点紧密相关，如图 6.4.23 所示，电路中 JK 触发器具有上升沿触发的动作特点，为二进制减法异步计数器，若采用具有下降沿触发动作特点的 JK 触发器，则构成二进制加法异步计数器。由下降沿 JK 触发器构成的 4 位二

进制加法计数器如图 6.4.30 所示。

通过上面分析可以看出，异步计数器存在过渡状态，响应速度慢，可靠性差。

当然，异步计数器也有优点，如图 6.4.23 所示的 3 位减法计数器只需 3 个触发器，无须门电路，结构简单。

此外，异步计数器使用更灵活，如在集成芯片内部电路中，断开如图 6.4.23 所示 3 位减法计数器触发器 1 与 2 的连接，作为外部引脚，具体如图 6.4.31 所示。

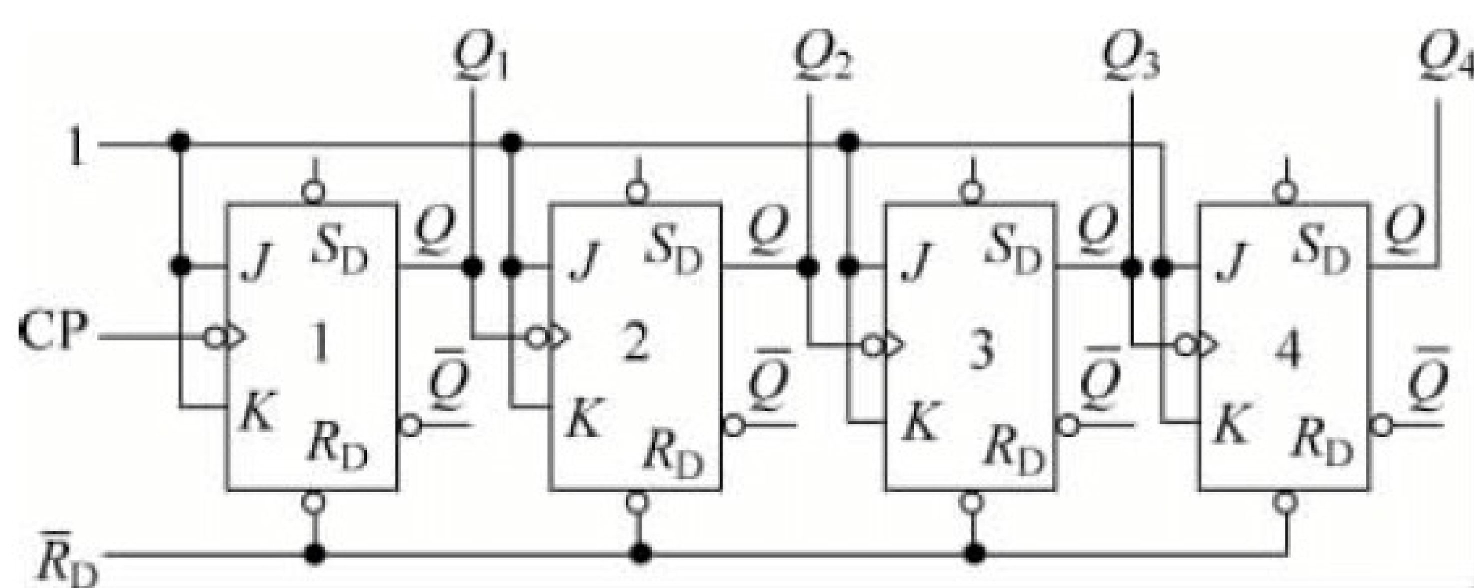


图 6.4.30　异步 4 位二进制加法计数器

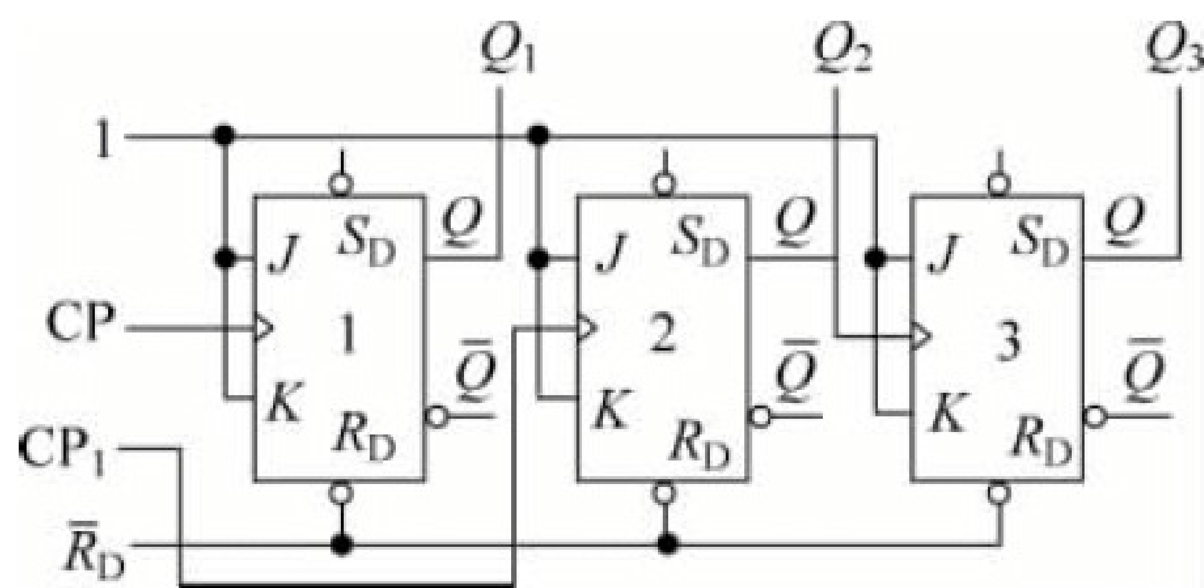


图 6.4.31　异步二—四—八进制计数器

在如图 6.4.31 所示电路中，触发器 1 为一个二进制计数器，触发器 2、3 构成一个四进制计数器。若从外部将外部引脚 Q_1、CP_1 重新短接，则如图 6.4.31 所示电路整体构成一个八进制计数器。

集成芯片常采用下降沿触发的触发器构成异步计数器。类似地，断开如图 6.4.30 所示 4 位二进制加法计数器电路的第 1 级触发器与第 2 个触发器之间的连接，将该连接端作为外部引脚引出，则该电路构成一个二—八—十六进制计数器。

6.4.5　集成异步计数器

1. 集成二进制异步计数器

集成二进制异步计数器的品种较多。以比较典型的芯片 74LS197 4 位二进制异步加法计数器为例作如下说明。

74LS197 是在图 6.4.30 的基础上，为了使用和扩展功能的方便，在制作集成 4 位二进制异步加法计数器时，增加了一些辅助功能。

(1) 引脚说明。如图 6.4.32 所示为异步 4 位二进制计数器 74LS197。图中，$\overline{CR}$为异步清零端，$CT/\overline{LD}$为计数和置数控制端，CP_0为触发器 0 的时钟脉冲输入端，CP_1为触发器 1 的时钟脉冲输入端，$D_0 \sim D_3$为并行数据输入端；$Q_0 \sim Q_3$为计数器的状态输出端。

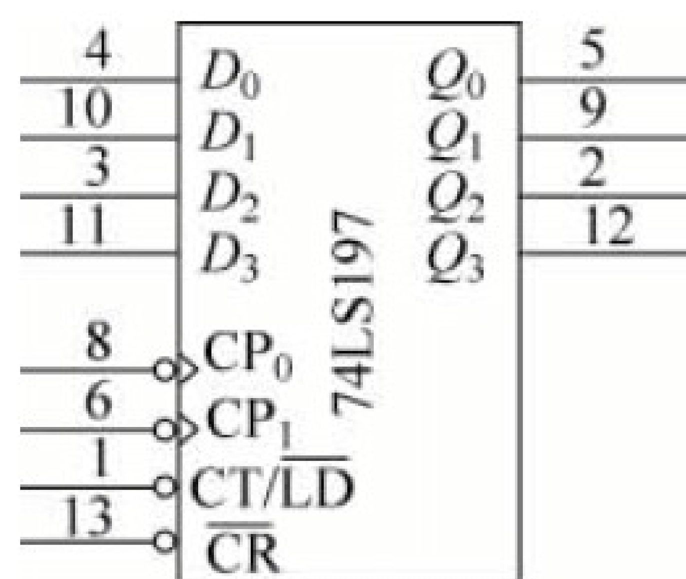


图 6.4.32　74LS197

(2) 功能表。如表 6.4.7 所示为 74LS197 集成 4 位二进制异步计数器功能表。由表 6.4.7 可以清楚地看出，74LS197 具有以下功能：

① 清零功能。当$\overline{CR}=0$时，计数器异步清零。

表 6.4.7　74LS197 功能表

$\overline{CR}$	$CT/\overline{LD}$	CP	$D_0\ D_1\ D_2\ D_3$	$Q_0\ Q_1\ Q_2\ Q_3$
0	×	×	× × × ×	0 0 0 0
1	0	×	$d_0\ d_1\ d_2\ d_3$	$d_0\ d_1\ d_2\ d_3$
1	1	↓	× × × ×	加法计数

② 置数功能。当$\overline{CR}$=1，$CT/\overline{LD}$=0 时，计数器异步置数。

③ 4 位二进制异步加法计数功能。当$\overline{CR}$=1，$CT/\overline{LD}$=1 时，进行异步加法计数。

应注意：若将 CP 加在 CP_0端、把 Q_0与 CP_1连接起来，则构成 4 位二进制即十六进制加法计数器，若将 CP 加在 CP_1端，则计数器中的触发器 1、2、3 构成 3 位二进制即八进制异步加法计数器，显然触发器 0 不工作。若只将 CP 加在 CP_0端，CP_1端接 0 或 1，则只有触发器 0 工作，为 1 位二进制计数器。此时触发器 1、2、3 均不工作。所以，也常将 CT54197/CT74LS197 称为二—八—十六进制异步计数器。

与 74LS197 二—八—十六进制异步计数器相同的芯片有 74LS293。双 4 位二进制异步加法计数器的芯片有 74LS393。CMOS 集成异步计数器有 7 位二进制异步计数器 CC4024，12 位二进制异步计数器 CC4040，14 位二进制异步计数器 CC4020、CC4060 等。

2. 集成十进制异步计数器



常用的集成十进制异步计数器型号有 74LS196、74LS290 等，它们都是按照 8421BCD 码进行加法计数的电路。下面以比较典型的 74LS290 为例进行简单说明。

(1) 引脚说明。如图 6.4.33 所示为异步十进制计数器 74LS290。图中 S_{9A}、S_{9B}为异步置“9”输入端，R_{0A}、R_{0B}为异步清零输入端，CP_0为触发器 0 的时钟脉冲输入端，CP_1为触发器 1 的时钟脉冲输入端；Q_0～Q_3为计数器的状态输出端。

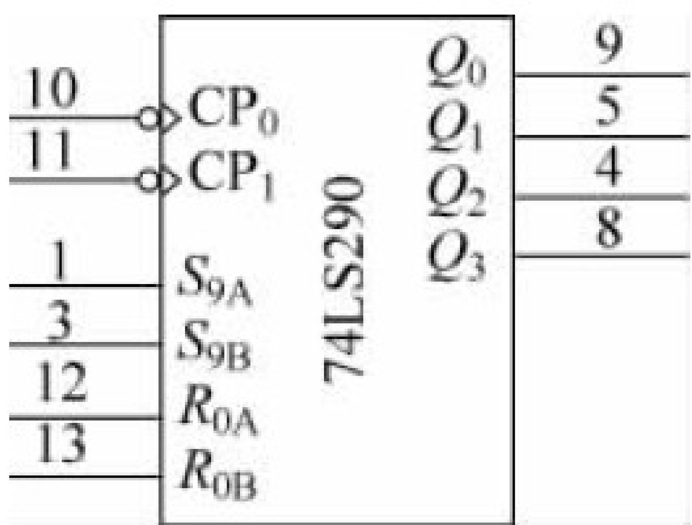


图 6.4.33　74LS290

(2) 功能表。如表 6.4.8 所示为集成十进制异步计数器 74LS290 的功能表。由表 6.4.8 可以清楚地看出，74LS290 具有以下功能：

表 6.4.8　74LS290 功能表

R_{0A}	R_{0B}	S_{9A}	S_{9B}	CP	$Q_0\ Q_1\ Q_2\ Q_3$
1	1	0	×	×	0 0 0 0
1	1	×	0	×	0 0 0 0
×	×	1	1	×	1 0 0 1
×	0	×	0	↓	加法计数
×	0	0	×	↓	加法计数
0	×	×	0	↓	加法计数
0	×	0	×	↓	加法计数

① 异步清零功能。当 $R_{0A}=R_{0B}=1$,$S_{9A}\cdot S_{9B}=0$ 时,计数器异步清零。

② 置“9”功能。当 $S_{9A}=S_{9B}=1$ 时,计数器实现置“9”功能,即被置 1001 状态。显然,这种置“9”也是通过触发器输入端进行的,与 CP 脉冲无关,而且优先级别高于 R_{0A}、R_{0B}。

③ 计数功能。74LS290 具有典型异步计数器的特点,具有两个外部时钟输入端,包括一个二进制计数器和一个五进制计数器。有 4 种基本情况:

若将外部输入的计数脉冲 CP→CP_0,则二进制计数器工作;若从外部连接 Q_0→CP_1,五进制计数器工作,对二进制计数器溢出信号 Q_0↓计数,则电路将对 CP 按照 8421BCD 码进行异步加法计数,为十进制自然规律加法计数器。

若仅将输入的计数脉冲 CP 接到 CP_0端,即 CP_0=CP,而 CP_1与 Q_0不连接,CP_1悬空,则二进制计数器工作,五进制计数器无时钟,不工作,为二进制计数器。

若只将 CP 计数脉冲接在 CP_1端,即 CP_1=CP,CP_0悬空,则二进制计数器无时钟,不工作,五进制计数器工作,为五进制异步计数器。

若 CP→CP_1,则五进制计数器工作;Q_3→CP_0,二进制计数器也工作,对五进制计数器溢出信号 Q_3↓计数,电路仍然是十进制异步计数器,但计数规律就不再是按照 8421BCD 码计数了。此时的计算机仿真结果如图 6.4.34 所示。

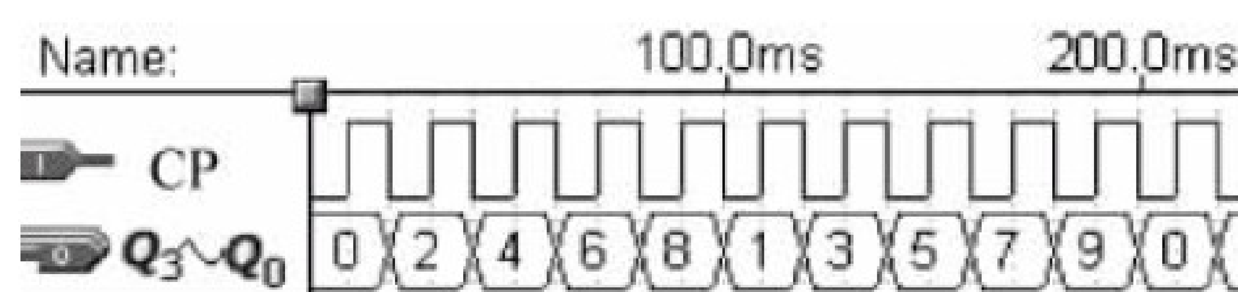


图 6.4.34　CP_1=CP,CP_0=Q_3时的仿真图

可这样理解上面的状态变化规律。依照接法,二进制计数器对五进制计数器输出 Q_3 的下降沿计数。五进制计数器计数到“4”时 $Q_3=1$,溢出时计数器 4→0,Q_3输出下降沿,因此,在该接法下,二进制计数器对五进制计数器的溢出信号计数。

CP_1=CP,先进行五进制计数,二进制计数器输出 Q_0保持 0 不变,五个状态变化规律为

$$0\rightarrow2\rightarrow4\rightarrow6\rightarrow8$$

之后,五进制计数器溢出,二进制计数器加 1,$Q_0=1$,五进制计数器继续计数,5 个状态变化规律为

$$1\rightarrow3\rightarrow5\rightarrow7\rightarrow9$$

在 Quartus Ⅱ中,74LS290 原理电路如图 6.4.35 所示。其中,CLKA 对应 CP_0,CLKB 对应 CP_1。SET9A、SET9B 对应 S_{9A}、S_{9B},CLRA、CLRB 对应 R_{0A}、R_{0B},QA~QD 对应 Q_0~Q_3。通过该原理电路不难看出,SET9A、SET9B 的异步置“9”功能的实现是通过直接设置 D 触发器 QA、QD 的异步置数端及 QB、QC 的异步清零端实现的,有兴趣的读者可参考该电路进一步理解 74LS290 的逻辑功能。

【例 6.4.3】 电路如图 6.4.36 所示,请分析该电路为几进制计数器。

解　在如图 6.4.36 所示电路中,CP→CP_0,Q_0→CP_1,五进制计数器工作,整体构成十进制自然规律加法计数器。

图 6.4.36 中,置“9”输入端无效,Q_3→R_{0A}、R_{0B},因此,当 $Q_3=1$ 时,$R_{0A}=R_{0B}=1$,计数

图 6.4.35 74LS290 的电路原理图

器异步清零。即当计数器 $Q_3Q_2Q_1Q_0$ 计数到“0111”之后，再来一个脉冲进入状态“1000”，此时，$Q_3=1$，计数器立即回到“0000”。

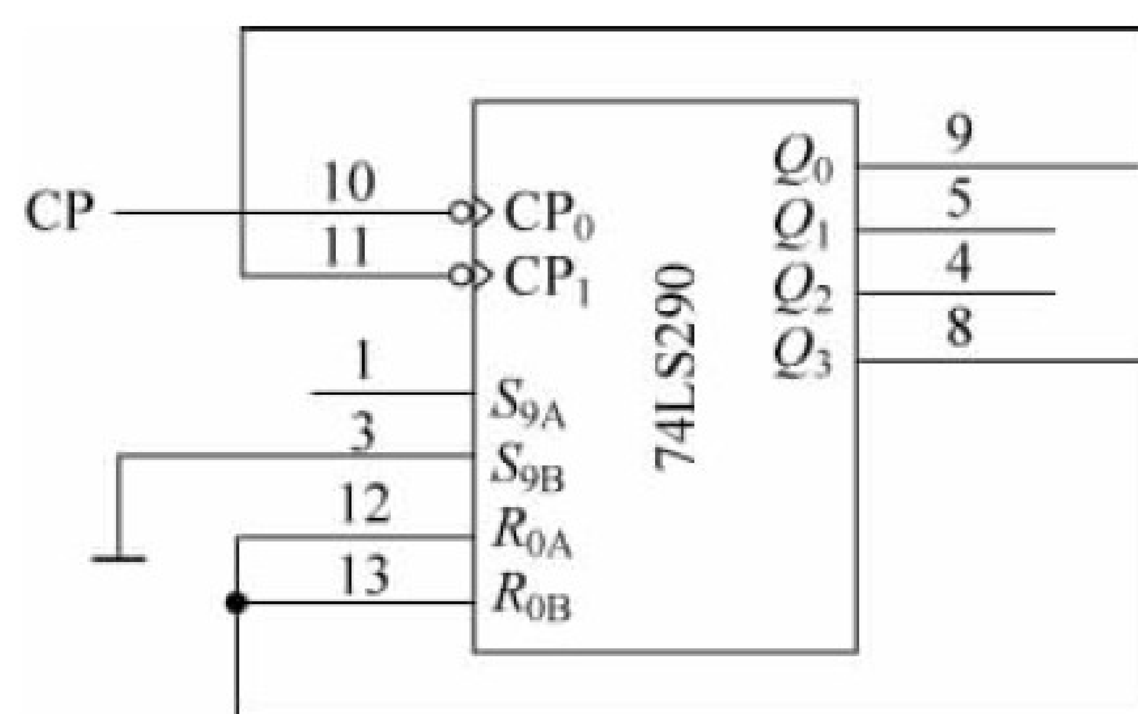


图 6.4.36 例 6.4.3 的图 1

可见，如图 6.4.36 所示电路经 8 个时钟，历“0→…→7→8”9 个状态后回到 0(其中 7→8→0 只需要一个时钟)，为八进制异步加法计数器。

如图 6.4.36 所示电路在 Quartus Ⅱ 中的仿真结果如图 6.4.37 所示。仿真结果显示该电路为八进制加法计数器。

由图 6.4.37 可清晰看出，如图 6.4.36 所示电路为下降沿的加法计数器。

6.4.5 小节指出，异步计数器存在过渡状态，响应速度慢。如图 6.4.37 所示仿真结果似乎表明如图 6.4.36 所示电路状态变化较为同步，没有明显的过渡状态。

必须指出，异步电路的过渡状态持续时间极短，在如图 6.4.37 所示仿真图中，脉冲宽度

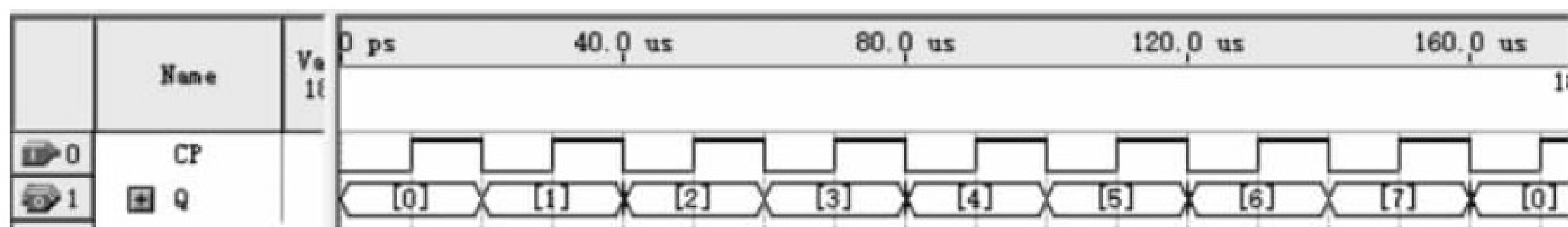


图 6.4.37 例 6.4.3 的图 2

为 10μs，远远大于过渡状态的持续时间，因此，无法看出明显的过渡状态。

可设置输入时钟 CP 的脉冲宽度为 50ns，仿真结果如图 6.4.38 所示。由该仿真图可明显看出异步计数器存在过渡状态。其中，状态“7→0”之间的放大图如图 6.4.39 所示。可见，状态“7→0”之间包括过渡状态的状态变化过程为“7→6→4→0 →8→0 ”，包括 4 个中间状态。

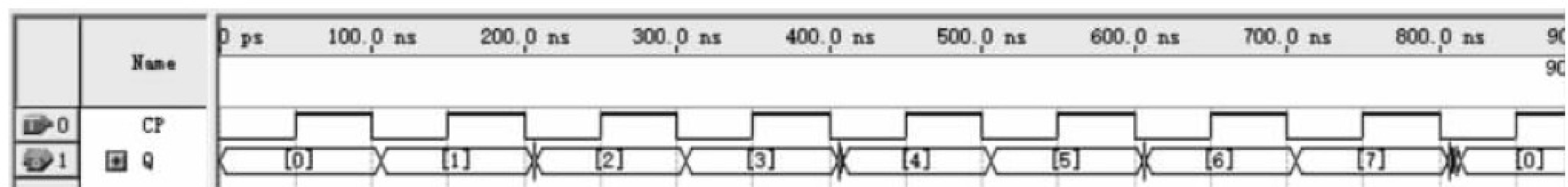


图 6.4.38 例 6.4.3 的图 3

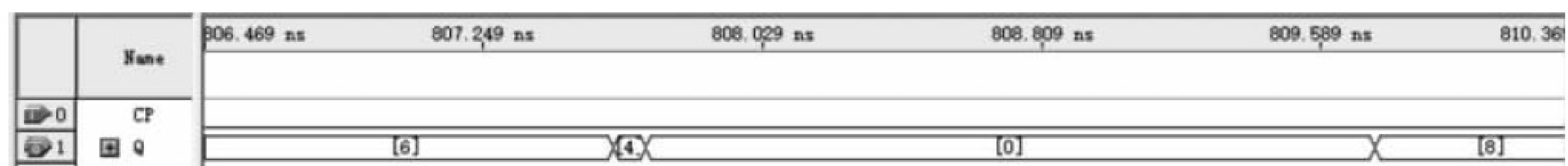


图 6.4.39 例 6.4.3 的图 4

6.4.6 利用集成同步计数器实现 N 进制计数器

集成的计数器芯片主要包括二进制计数器、十进制计数器两大类。应用实践中的计数器往往不是单纯的二进制或十进制计数器，而是各种需求的 N 进制计数器。

1. N 进制计数器设计的一般方法



可利用集成的 M 进制计数器芯片实现 N 进制计数器。当 $M>N$ 时，利用集成计数器的清零控制端或者置数控制端，使设计的电路跳过某些状态而获得 N 进制计数器。

集成计数器清零、置数有异步和同步两种工作方式。所谓异步工作方式，是指通过时钟触发器异步输入端（$\bar{R}_D$ 端或 $\bar{S}_D$）实现清零或置数，而与 CP 计数脉冲无关。同步工作方式是指当 CP 计数脉冲到来时，才能完成清零或者置数的任务。

可利用支持同步清零的 M 进制计数器芯片实现 N 进制计数器。设计思路如图 6.4.40 所示。

如图 6.4.40 所示的 M 进制计数器芯片具有 M 个状态。可利用集成计数器的同步清零控制端使循环过程中设法跳过 $M-N$ 个状态可实现 N 进制计数器。计数器从 S_0 开始计数，$N-1$ 个脉冲后计数器进入状态 S_{N-1}；此时，令同步清零端有效；当下个时钟到来时，计数器回到 S_0。可见，M 进制计数器经 N 个时钟，从状态 S_0 又回到 S_0，为 N 进制计数器。

也可利用支持异步清零的 M 进制计数器芯片实现 N 进制计数器。设计思路如图 6.4.41 所示。

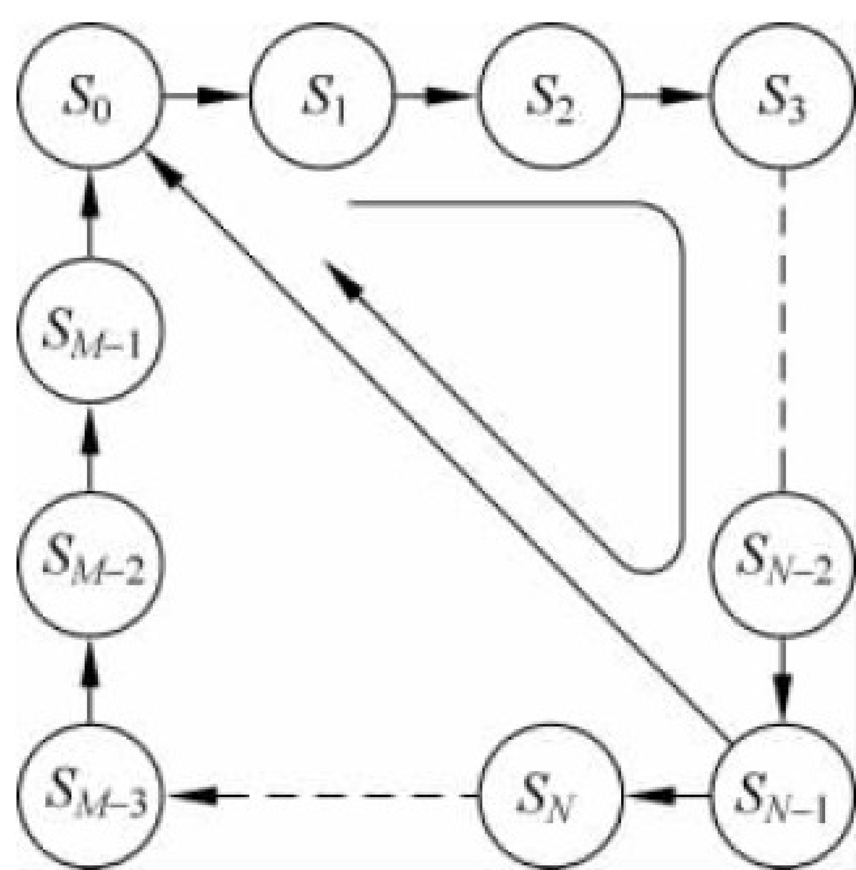


图 6.4.40 N 进制计数器的图 1

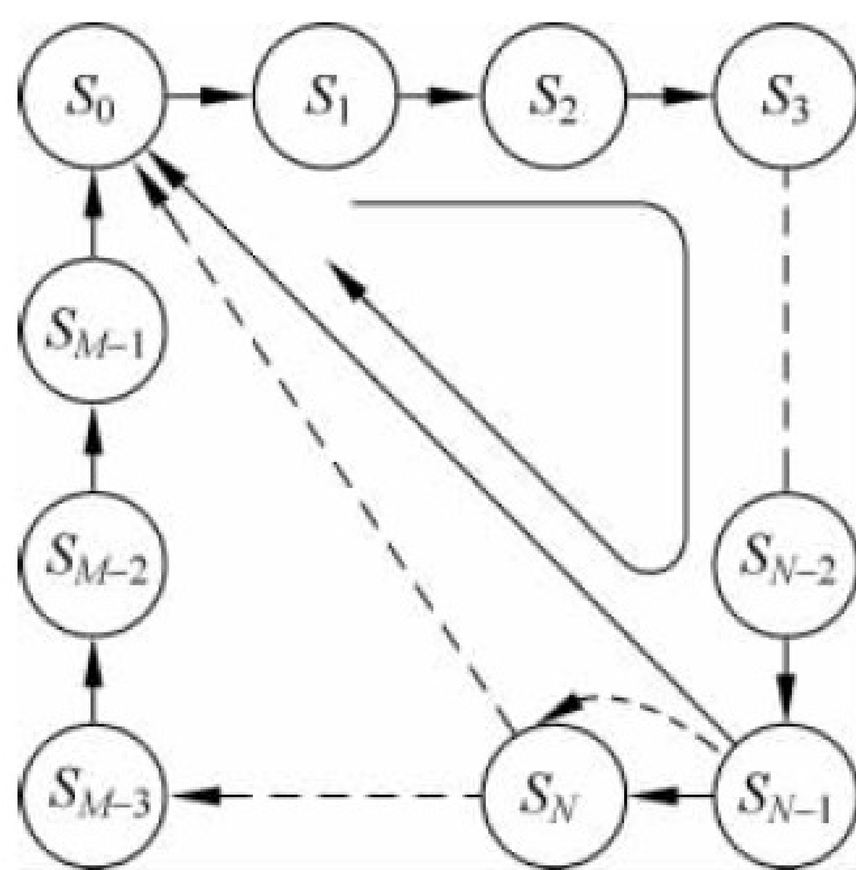


图 6.4.41 N 进制计数器的图 2

如图 6.4.41 所示的 M 进制计数器芯片具有 M 个状态。可利用集成计数器的异步清零控制端使循环过程中设法跳过 $M-N-1$ 个状态可实现 N 进制计数器。计数器从 S_0 开始计数，N 个脉冲后计数器进入状态 S_N；此时，令异步清零端有效，计数器立即回到 S_0。可见，M 进制计数器经 N 个时钟，从状态 S_0 又回到 S_0。因此，尽管 M 进制计数器经历了 $N+1$ 个状态，但只经历了 N 个时钟，所以依旧为 N 进制计数器。

可利用支持同步置数的 M 进制计数器芯片实现 N 进制计数器。设计思路如图 6.4.42 所示。

如图 6.4.42 所示的 M 进制计数器芯片具有 M 个状态。可利用集成计数器的同步置数控制端使循环过程中设法跳过 $M-N$ 个状态可实现 N 进制计数器。计数器从 S_0 开始计数，i 个脉冲后计数器进入状态 S_i；此时，使同步置数端有效，下个时钟到来时，进入预置状态 S_j，之后，继续计数直到计数器回到 S_0。如果 $j-i=M-N+1$，则计数器跳过了 $M-N$ 个状态，可见，M 进制计数器经 N 个时钟，从状态 S_0 又回到 S_0，为 N 进制计数器。

若 $i=N-1$，$j=0$，则状态图如图 6.4.40 所示，与同步清零相同。

还可利用支持异步置数的 M 进制计数器芯片实现 N 进制计数器。设计思路如图 6.4.43 所示。

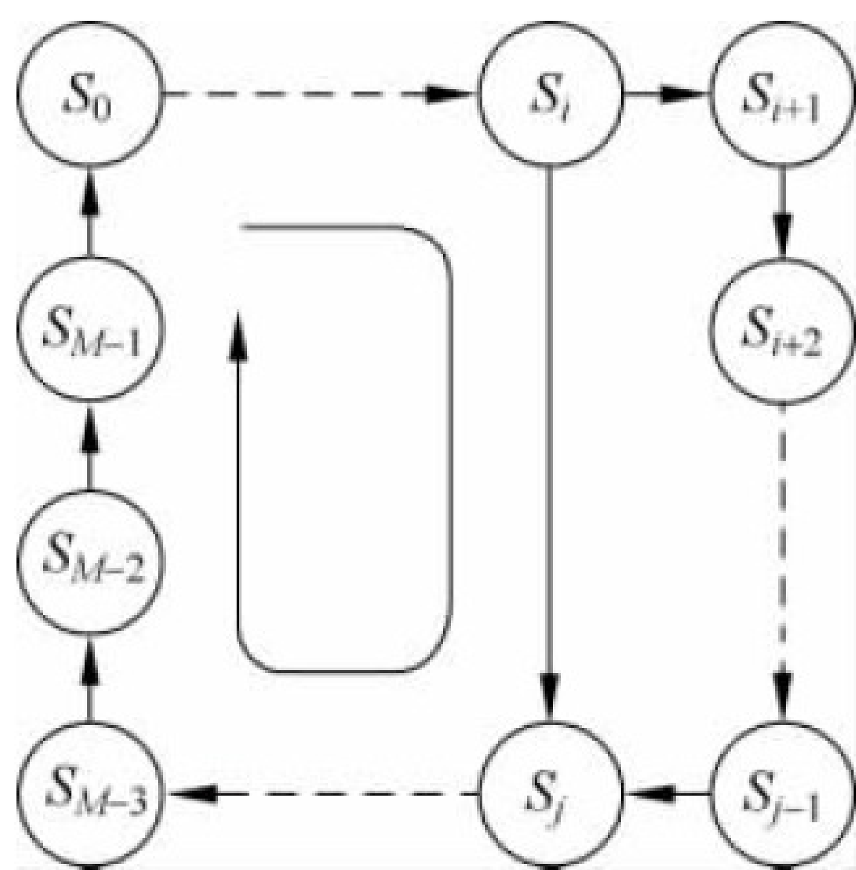


图 6.4.42 N 进制计数器的图 3

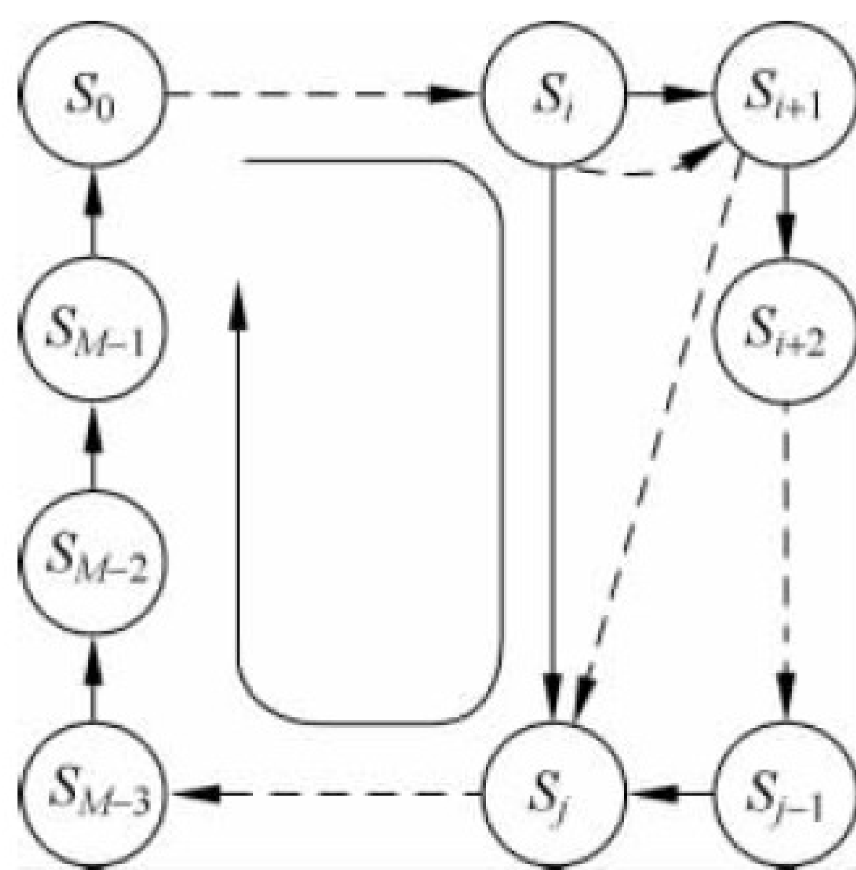


图 6.4.43 N 进制计数器的图 4

如图 6.4.43 所示的 M 进制计数器芯片具有 M 个状态。可利用集成计数器的异步置数控制端使循环过程中设法跳过 $M-N-1$ 个状态可实现 N 进制计数器。计数器从 S_0 开始计数，进入状态 S_{i+1} 后使异步置数端有效，立即进入预置状态 S_j，之后，继续计数直到计数器回到 S_0。如果 $j-i=M-N+1$，则计数器跳过了 $M-N-1$ 个状态，可见，M 进制计数器经 N 个时钟，从状态 S_0 又回到 S_0，为 N 进制计数器。

设初始置数状态为 i，若 $i=N$，$j=0$，则状态图如图 6.4.41 所示，与异步清零相同。

2. 用同步清零端复位实现 *N* 进制计数器



用同步清零端复位实现 N 进制计数器的思路如图 6.4.40 所示。具体而言，M 进制计数器从状态 S_0 开始计数时，$N-1$ 个脉冲后，M 进制计数器处于 S_{N-1} 状态。利用 S_{N-1} 状态产生一个清零信号，加到同步清零输入端，下个时钟到来时，计数器回到状态 S_0，从而跳过了 $M-N$ 个状态，实现了模值数为 N 的 N 进制计数器。

用同步清零端复位实现 N 进制计数器的步骤如下：

(1) 选择具有同步清零功能的 M 进制计数器芯片型号($M>N$)。

(2) 写出状态 S_{N-1} 的二进制代码。

(3) 求出清零函数。

(4) 画出最终电路图。

【例 6.4.4】 请用同步清零的复位法设计一个六进制计数器。

解 (1) 确定计数器芯片型号。由表 6.4.3 知，支持同步清零功能的计数器有 74LS162、74LS163，均满足 $M>N$，选用 74LS163 实现本例。

(2) 写出计数器 $Q_3Q_2Q_1Q_0$ 状态 S_{N-1} 的二进制代码，即

$$S_{N-1} = S_5 = 0101$$

(3) 求出清零函数 $\overline{\text{CR}}$。当然，74LS163 具有 16 个状态，还有状态 6～15 等 10 个状态未使用，为无关项，考虑无关项，求清零函数时只考虑等于 1 的量即可，有

$$\overline{\text{CR}} = \overline{Q_2\, Q_0} \tag{6.4.7}$$

(4) 画出电路如图 6.4.44 所示。注意，具体绘制电路时，除清零函数、外部时钟外，应将计数允许接为有效($\text{CT}_\text{P}=\text{CT}_\text{T}=1$)。

(5) 计算机仿真分析。如图 6.4.44 所示电路在 Quartus Ⅱ 中的仿真结果如图 6.4.45 所示。图中，out 对应清零函数 $\overline{\text{CR}}$。

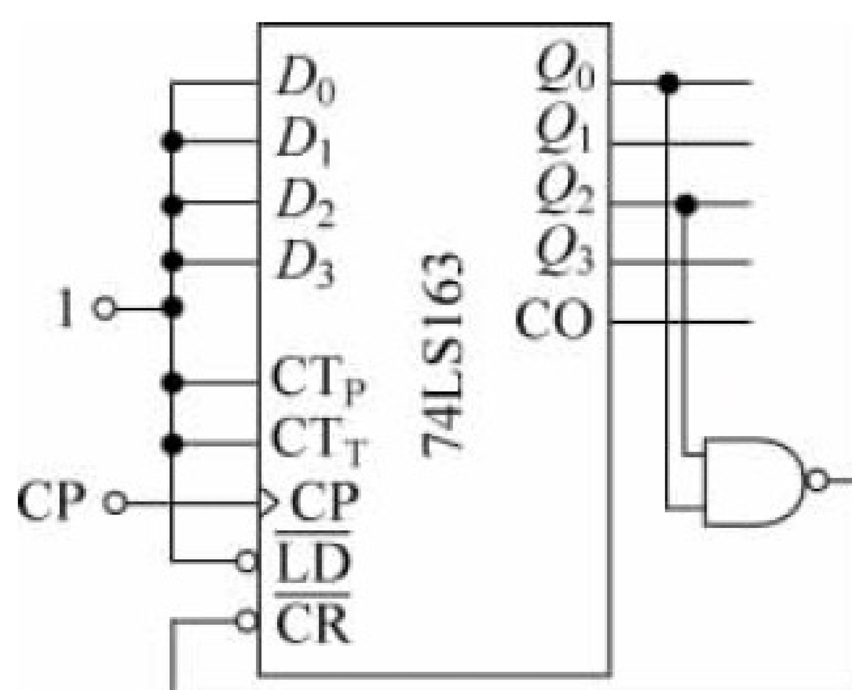


图 6.4.44 例 6.4.4 的图 1

由仿真结果可看出，当计数器 $Q_3Q_2Q_1Q_0$ 进入状态“5”时，清零信号有效，下个时钟到来时，计数器回到 0，为一个自然规律计数的六进制计数器。由仿真结果还可看出，计数器溢出时，清零函数 $\overline{\text{CR}}$ 输出上升沿，脉冲宽度为时钟信号宽度的 2 倍，可作为进位信号使用。

类似地，在如图 6.4.46 所示电路中，计数允许信号一直有效，清零函数$\overline{CR}=\overline{Q_3Q_1Q_0}$。即当计数器 $Q_3Q_2Q_1Q_0$进入状态“1011”时，清零信号有效，下个时钟到来时，计数器回到 0。计数器状态变化规律如下：

$$0 \to 1 \to \cdots \to 11 \to 0$$

可见，如图 6.4.46 所示电路构成了一个十二进制计数器。

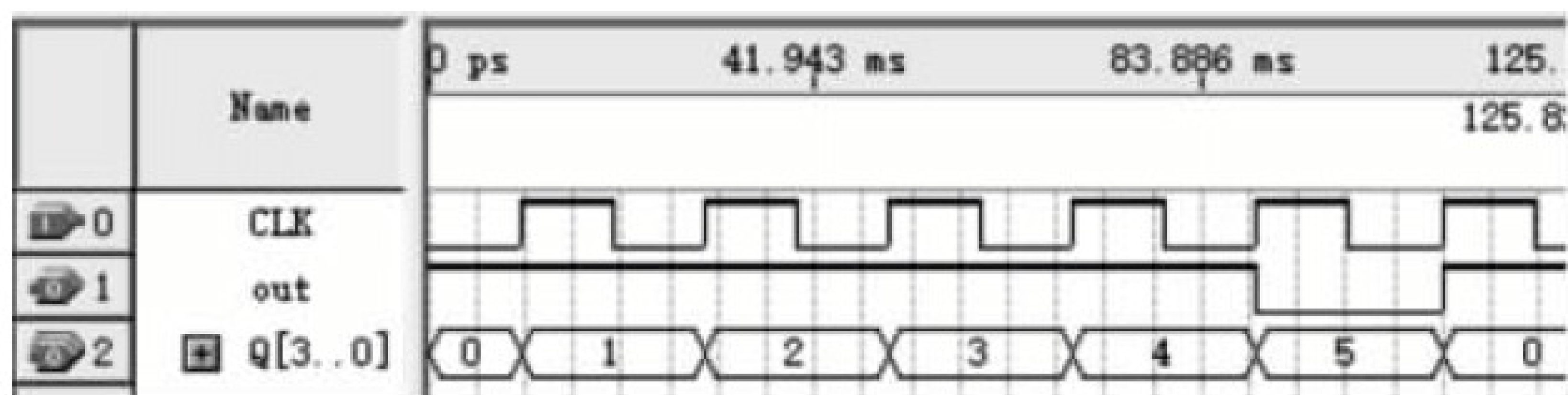


图 6.4.45 例 6.4.4 的图 2

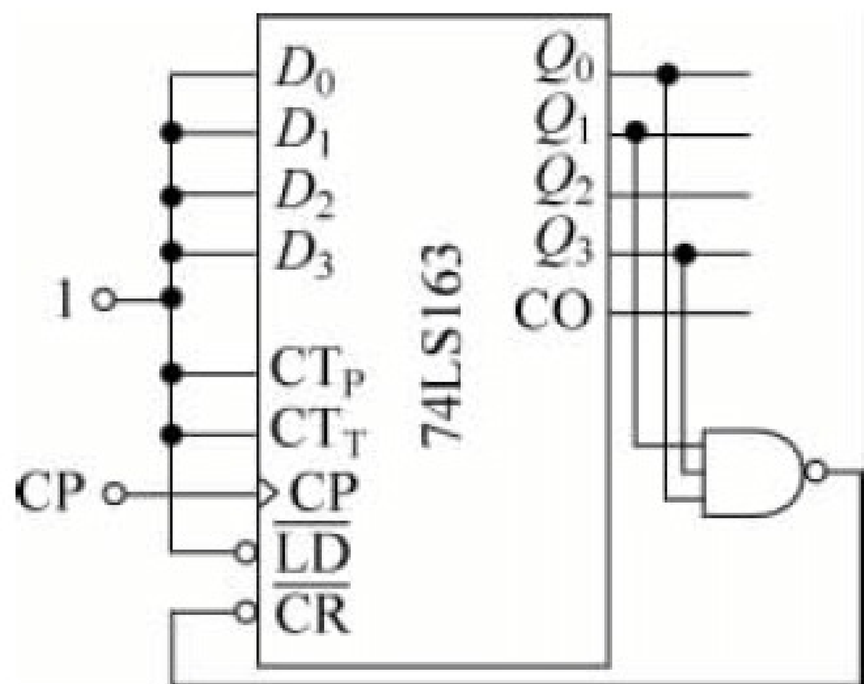


图 6.4.46 十二进制计数器

3. 用异步清零端复位实现 *N* 进制计数器

用异步清零端复位实现 N 进制计数器的思路如图 6.4.41 所示。M 进制计数器从状态 S_0开始计数时，N 个脉冲后，M 进制集成计数器处于 S_N状态；利用状态 S_N产生一个清零信号，加到异步清零输入端，则使计数器回到状态 S_0，这样就跳过了 $M-N-1$ 个状态，实现了模值数为 N 的 N 进制计数器。

设计步骤如下：

(1) 选择具有异步清零功能的 M 进制计数器芯片型号($M>N$)。

(2) 写出状态 S_N的二进制代码。

(3) 求出清零函数。

(4) 画出电路图。

【例 6.4.5】 请用异步清零的复位法设计一个六进制计数器。

解 (1)确定计数器芯片型号。由表 6.4.3 知，支持异步清零功能的计数器有 74LS160、74LS161，均满足 $M>N$，选用 74LS160 实现本例。

(2) 写出状态 S_N的二进制代码，即

$$S_N = S_6 = 0110$$

(3) 求出清零函数$\overline{CR}$。考虑无关项，求清零函数时只考虑等于 1 的量即可，有

$$\overline{CR} = \overline{Q_2 Q_1} \tag{6.4.8}$$

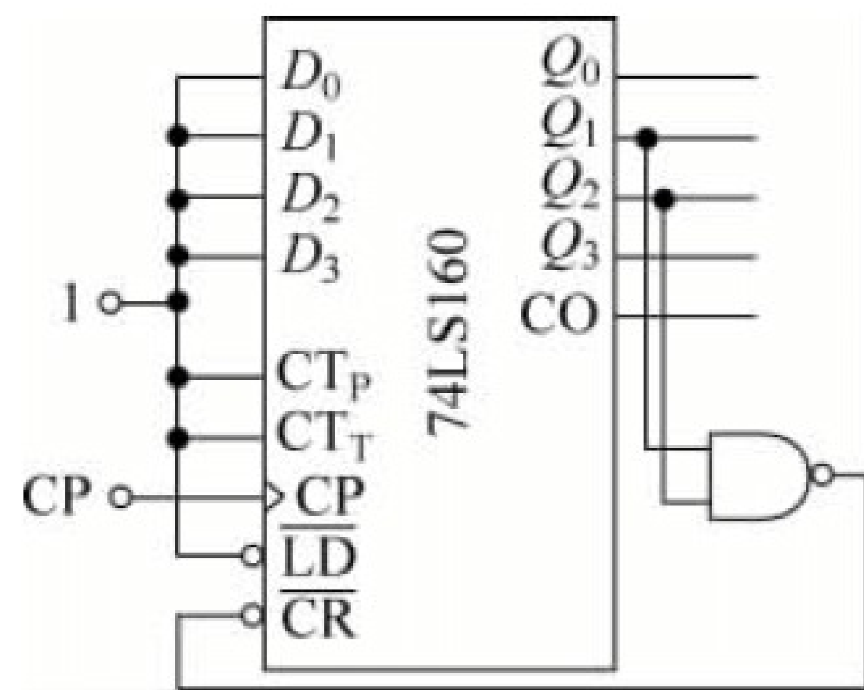


图 6.4.47 例 6.4.5 的图 1

(4) 画出电路如图 6.4.47 所示，图中，计数允许 $CT_P=CT_T=1$，置数允许接为无效。

(5) 计算机仿真。如图 6.4.47 所示电路在 Quartus Ⅱ

中的仿真结果如图 6.4.48 所示。图中，out 对应清零函数$\overline{CR}$。状态“5”～“0”之间的局部放大图如图 6.4.49 所示。

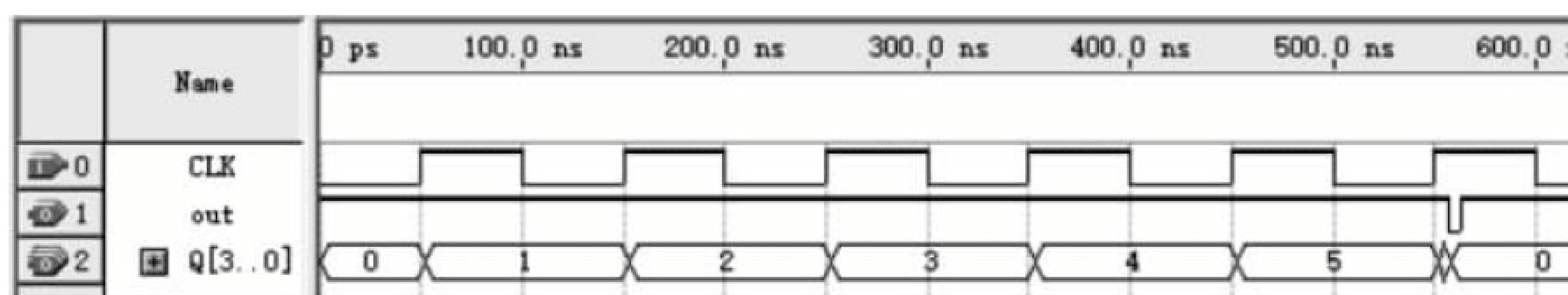


图 6.4.48　例 6.4.5 的图 2

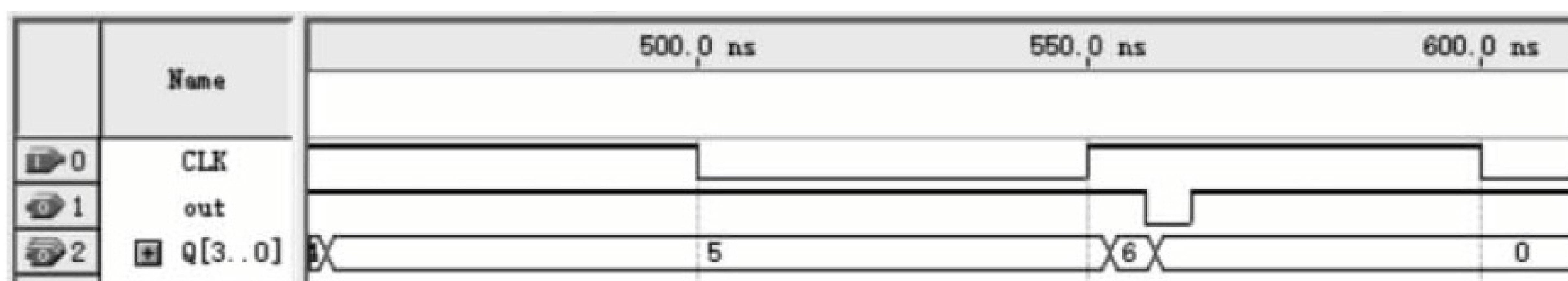


图 6.4.49　例 6.4.5 的图 3

由图 6.4.49 可看出，当计数器 $Q_3Q_2Q_1Q_0$进入状态“6”时，清零信号有效，之后，计数器立即回到 0。因为计数器进入状态“5”后，在一个时钟的作用下，计数器由“5”跳变到“6”后很快到“0”，因此，尽管计数器经历了 6 个状态，但脉冲的个数只有 5 个，所以为五进制计数器。

由仿真结果还可看出，计数器溢出时，清零函数$\overline{CR}$输出上升沿，具有进位信号的特点。但$\overline{CR}$脉冲宽度持续时间太短，只有几纳秒，直接作为进位信号使用时，可靠性差。

类似地，在如图 6.4.50 所示电路中，计数允许信号一直有效，清零函数$\overline{CR}=\overline{Q_3Q_2}$。即当计数器 $Q_3Q_2Q_1Q_0$进入状态“1100”时，清零信号有效，之后，计数器立即回到 0。计数器状态变化规律如下(其中，状态“1100”为过渡状态)：

$$0 \rightarrow 1 \rightarrow \cdots \rightarrow 11 \rightarrow 12(0)$$

可见，如图 6.4.50 所示电路构成了一个十二进制计数器。

从如图 6.4.47 和图 6.4.50 所示电路不难看出，利用异步清零端的复位法实现 N 进制计数，存在一个短暂的过渡状态 S_N。作为一个 N 进制计数器，从初始状态 S_0开始计数。当计到 S_{N-1}时，若输入一个 CP 计数脉冲，计数器的状态应该回到 S_0，同时向高一位产生进位输出信号，但是，用异步清零端的复位法所设计出的计数器，不是立即回到 S_0，而是先转换到 S_N状态，借助 S_N产生清零信号使计数器回到 S_0状态，这时状态 S_N消失，整个过程需要几十纳秒。

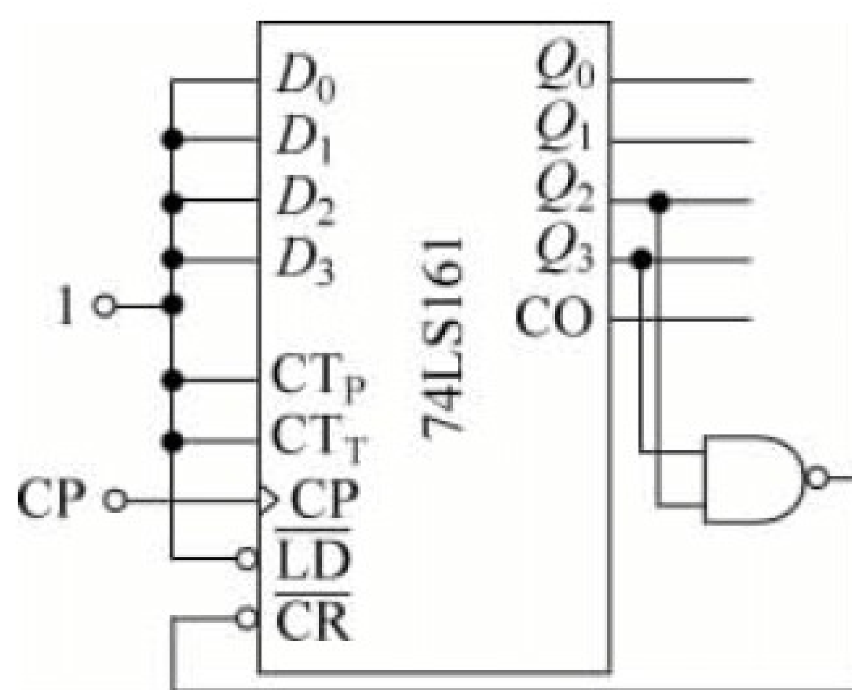


图 6.4.50　十二进制计数器

此外，清零函数不适合直接作为进位信号使用，如图 6.4.47 所示电路的改进电路如图 6.4.51 所示。

如图 6.4.51 所示电路在 Quartus Ⅱ 中的仿真结果如图 6.4.52 所示。图中，out 对应

进位信号 CO。清零函数为进位信号 CO 的反。

对照图 6.4.48 和图 6.4.52，如图 6.4.51 所示电路具有和如图 6.4.47 所示电路相同的逻辑功能。此外，out 信号波形和时钟信号波形大体相同，溢出输出上升沿，可作为进位信号使用。

下面简要分析如图 6.4.51 所示电路进位信号的工作原理。

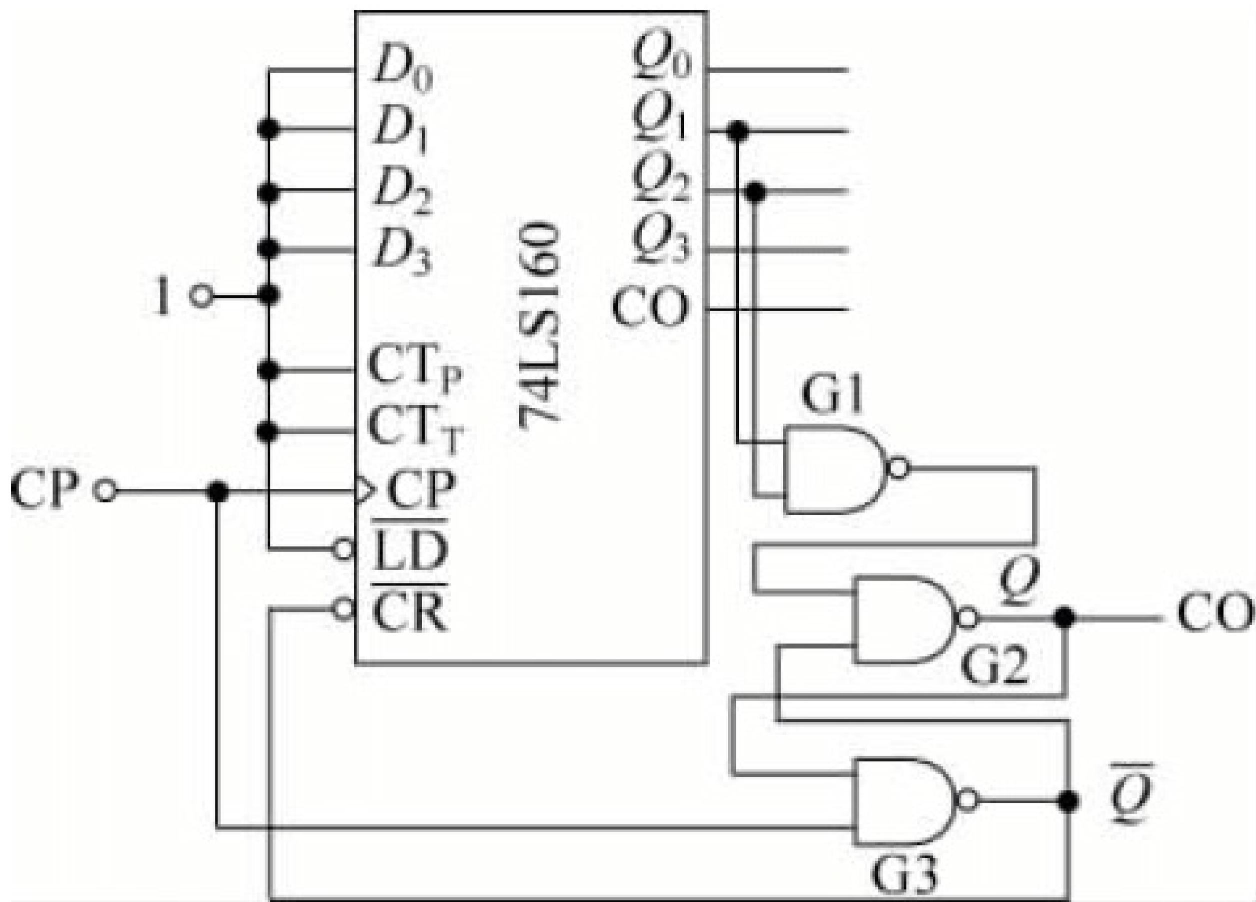


图 6.4.51　进位信号的实现

G2、G3 为用与非门构成的基本 RS 触发器。与非门 G1 的输出对应清零函数 $\overline{CR}=\overline{Q_1Q_2}$。当时钟信号上升沿到来，计数器 $Q_3Q_2Q_1Q_0$ 进入状态“0110”时，清零函数 $\overline{CR}$ 有效，为 0。此时，时钟信号为 1，基本 RS 触发器 $Q=1$ (CO=1)，$\overline{Q}=0$，计数器异步清零；计数器回到 0 后，清零函数 $\overline{CR}$ 失效，回到 1，但由于基本 RS 触发器的作用，保持 $Q=1$。

之后，CP 下降沿到来，时钟信号为 0，$\overline{Q}=1$；此时，清零函数早已失效，为 1，清除进位信号，$Q=0$，可求出进位信号波形，如图 6.4.52 倒数第 2 行所示。

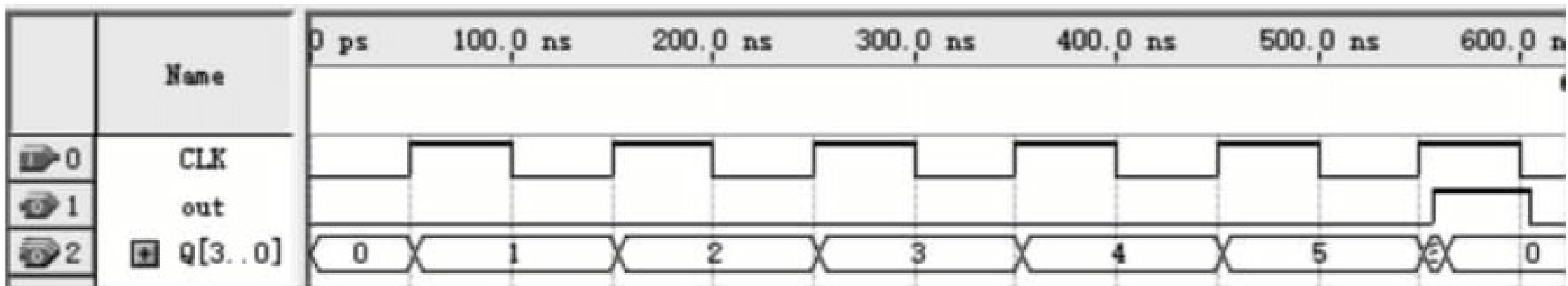


图 6.4.52　如图 6.4.51 所示电路的仿真图

4. 用同步置数端置数实现 N 进制计数器



用同步置数端置数实现 N 进制计数器主要有两种方式。

方式一：计数器从 S_0 开始计数，i 个脉冲后计数器进入状态 S_i；此时，使同步置数端有效，下个时钟到来时，进入预置状态 S_j，之后，继续计数直到计数器回到 S_0。使 i、j 关系满足 $j-i-1=M-N$，则计数器跳过了 $M-N$ 个状态，为 N 进制计数器，原理框图如图 6.4.42 所示。

方式二：使 $i=N-1$，$j=0$，同于同步清零，原理框图如图 6.4.40 所示。

方式一的好处是有一个集成芯片自身提供的进位信号，方式二的好处是实现的计数器为自然规律计数器。

本书中主要介绍采用方式二的思路实现 N 进制计数器。具体地讲，集成 M 进制计数器从状态 S_0 开始计数时，$N-1$ 个脉冲后，M 进制集成计数器处于 S_{N-1} 状态；利用 S_{N-1} 状态产生一个置数控制信号，加到置数控制端，当 CP 计数脉冲到来时，计数器回到状态 S_0，从而跳过了 $M-N$ 个状态，实现了模值数为 N 的 N 进制计数器。

具体设计步骤如下：

(1) 选择具有同步置数功能的 M 进制计数器芯片型号($M>N$)。

(2) 写出状态 S_{N-1} 的二进制代码。

(3) 求出置数函数 $\overline{\mathrm{LD}}$。

(4) 画出电路图。

【例 6.4.6】 请用同步置数的置数法设计一个六进制计数器。

解 (1) 确定计数器芯片型号。选用支持同步置数功能的 74LS163 实现本例。

(2) 写出计数器 $Q_3Q_2Q_1Q_0$ 状态 S_{N-1} 的二进制代码，即

$$S_{N-1}=S_5=0101$$

(3) 求出置数函数 $\overline{\mathrm{LD}}$。考虑无关项，求置数函数 $\overline{\mathrm{LD}}$ 时只考虑等于 1 的量即可，有

$$\overline{\mathrm{LD}}=\overline{Q_2\,Q_0} \tag{6.4.9}$$

(4) 画出电路，如图 6.4.53 所示。注意，具体绘制电路时，除置数函数、外部时钟外，应将计数允许接为有效($\mathrm{CT_P}=\mathrm{CT_T}=1$)。

(5) 计算机仿真分析。如图 6.4.53 所示电路在 Quartus Ⅱ 中的仿真结果如图 6.4.54 所示。图中，out 对应置数函数 $\overline{\mathrm{LD}}$。

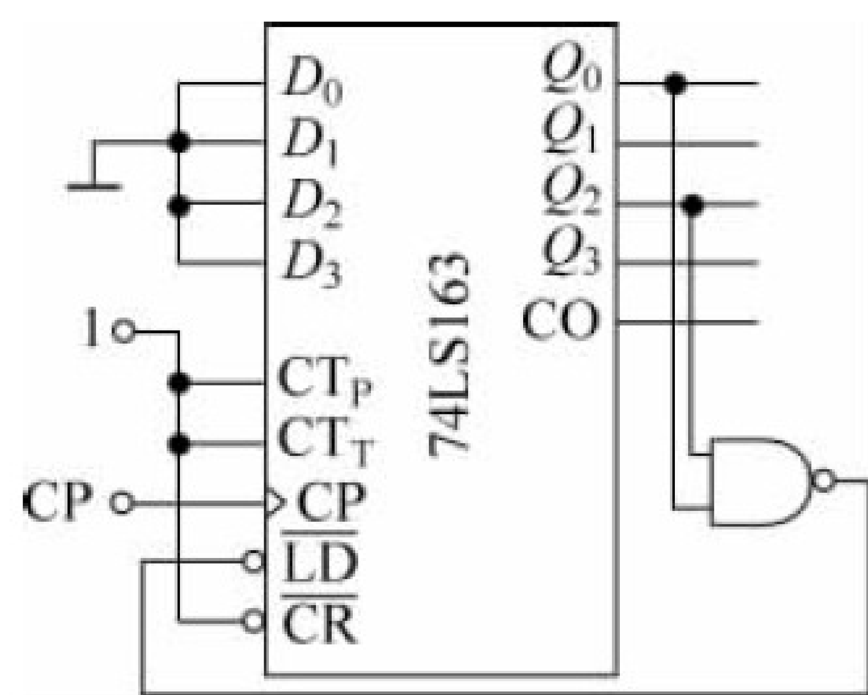


图 6.4.53 例 6.4.6 的图 1

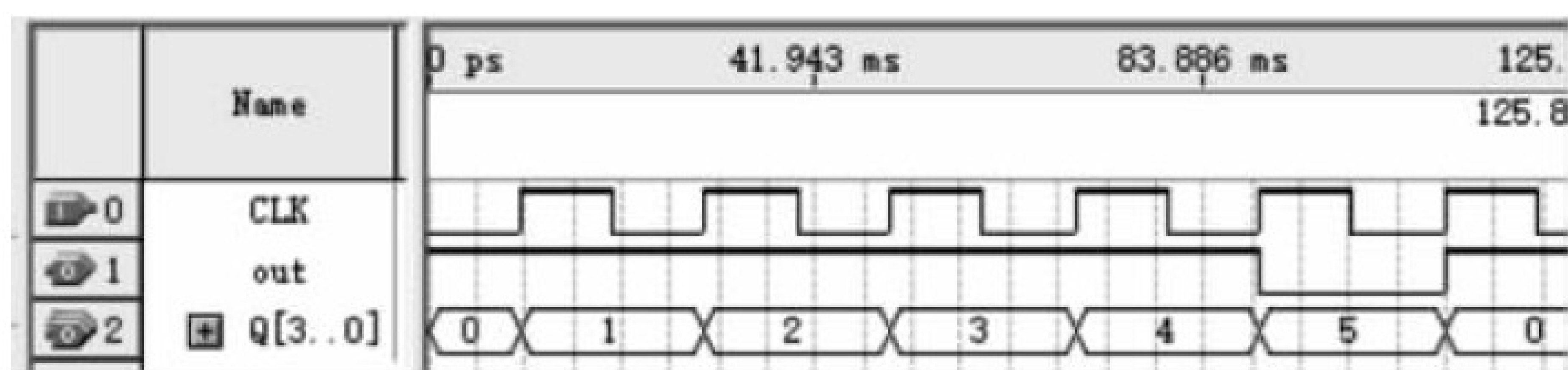


图 6.4.54 例 6.4.6 的图 2

由仿真结果可看出，当计数器 $Q_3Q_2Q_1Q_0$ 进入状态“5”时，置数信号有效，下个时钟到来时，计数器回到 0，为一个自然规律计数的六进制计数器。由仿真结果还可看出，计数器溢出时，置数函数输出上升沿，脉冲宽度为时钟信号宽度的 2 倍，可作为进位信号使用。

对照例 6.4.4 和例 6.4.6 不难发现，同步清零与同步置零设计思路、最终效果相同。

类似地，如图 6.4.55 所示电路中，计数允许信号一直有效，置数函数 $\overline{\mathrm{LD}}=\overline{Q_3Q_1Q_0}$。即当计数器 $Q_3Q_2Q_1Q_0$ 进入状态“1011”时，置数信号有效，有

$$N-1=1011_2=11$$

即 $N=12$。

可见，如图 6.4.55 所示电路构成十二进制计数器。

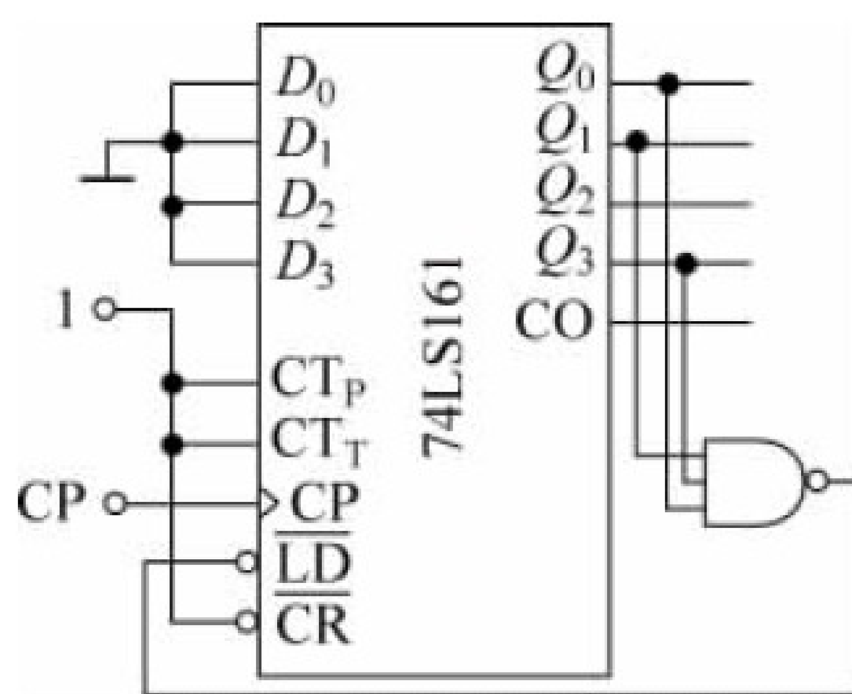


图 6.4.55 十二进制计数器

5. 用异步置数端置数实现 N 进制计数器

用异步置数端置数实现 N 进制计数器也有两种方式。

方式一：计数器从 S_0 开始计数，$i+1$ 个脉冲后计数器进入状态 S_{i+1}；此时，使异步置数端有效，之后，计数器立即进入预置状态 S_j，之后，继续计数直到计数器回到 S_0。使 i、j 的关系满足 $j-i=M-N-1$，则计数器跳过 $M-N-1$ 个状态，为 N 进制计数器，原理框图如图 6.4.43 所示。

方式二：使 $i=N$，$j=0$，同于异步清零，原理框图如图 6.4.41 所示。

本书中主要介绍采用方式二的思路实现 N 进制计数器，设计步骤如下：

(1) 选择具有异步置数功能的 M 进制计数器芯片型号($M>N$)。

(2) 写出状态 S_N 的二进制代码。

(3) 求出置数函数。

(4) 画出电路如图 6.4.56 所示。图中 $\overline{U}/D$，$\overline{CT}$接零。

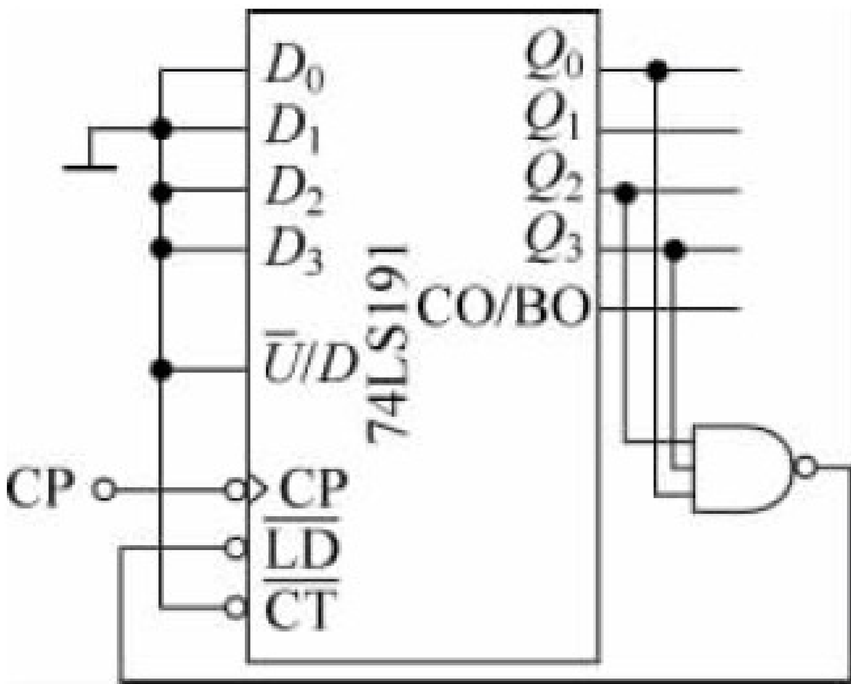


图 6.4.56 例 6.4.7 的图 1

【例 6.4.7】 请用异步置数的置数法设计一个十三进制计数器。

解 (1) 确定计数器芯片型号。由表 6.4.6 知，支持异步置数功能的单时钟计数器有 74LS190、74LS191 等，其中，74LS191 满足 $M>N$，选用 74LS191 实现本例。

(2) 写出状态 S_N 的二进制代码，即

$$S_N = S_{13} = 1101$$

(3) 求出置数函数。考虑无关项，求置数函数时只考虑等于 1 的量即可，有

$$\overline{LD} = \overline{Q_3 Q_2 Q_0} \tag{6.4.10}$$

(4) 画出电路如图 6.4.47 所示。图中，$\overline{U}/D$，$\overline{CT}$接零，当$\overline{LD}=1$ 时，CP 计数器脉冲到来，计数器加法计数；当第 13 个 CP 计数器脉冲到来时，计数器状态为“1101”，与非门输出低电平，$\overline{LD}=0$，并行异步置数，计数器很快翻转到“0000”，从而实现十三进制的加法计数。

(5) 计算机仿真。如图 6.4.56 所示电路在 QuartusⅡ中的仿真结果如图 6.4.57 所示。图中，out 对应置数函数$\overline{LD}$。

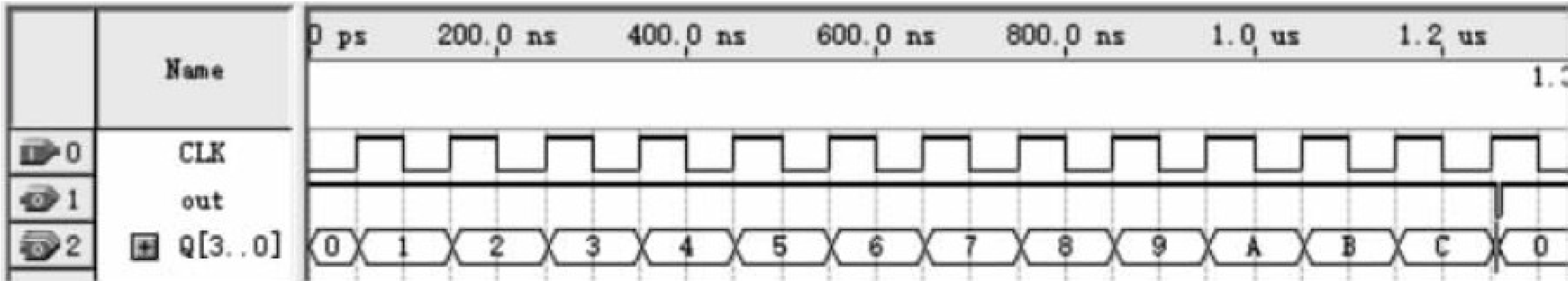


图 6.4.57 例 6.4.7 的图 2

对照例 6.4.5 和例 6.4.7 不难发现，异步清零与异步置零的设计思路、最终效果相同，置数函数$\overline{LD}$也不适合直接作为进位信号，可参考例 6.4.5 进一步改进如图 6.4.56 所示电

路，设计实现性能可靠的进位信号。

6.4.7 利用计数器的级联获得大容量 N 进制计数器



6.4.6 小节所介绍的用 M 进制计数器实现 N 进制计数器的方法均是针对 $N<M$ 的 N 进制计数器所进行的设计。如果需要设计的 N 进制计数器容量大于 M，则不可直接用单片 M 进制计数器实现 N 进制计数器，可将多片 M 进制计数器级联构成一个更大容量的计数器后再利用 6.4.6 小节介绍的方法实现 N 进制计数器。

1. 计数器级联的一般方法

集成的计数器芯片与计数直接相关的输入引脚主要有计数时钟、计数允许等；输出引脚有进位/借位输出，个别芯片提供了专门的级联端。

从级联计数角度，高位的计数器芯片完成的是对低位计数器芯片实际溢出的计数，级联后计数器的容量为高位计数器的值×低位计数器的值。如高位计数器容量为 5，低位计数器值为 7，高位计数器对低位计数器的溢出信号计数，低位计数器溢出输出上升沿，高位计数器上升沿计数，则级联后计数器的容量为 35。

从连接角度，计数器级联有两种方式：

(1) 高位计数器芯片直接对低位的进位/借位溢出信号计数，这种级联方式称为串行级联。

(2) 高位计数器芯片对时钟信号计数，利用低位的进位/借位溢出信号控制高位计数器的计数允许。只有进位/借位信号有效时，高位计数器才允许计数，这种级联方式称为并行级联。

2. 串行级联方法

串行级联实例如图 6.4.58 所示。图中，片 1、片 2 均为上升沿计数器。74LS160 为满幅输出上升沿、溢出输出下降沿的计数器，图中片 1 的溢出信号取反后接片 2 的时钟信号。两个芯片的计数允许、置数、清零等控制端相互短接，片 2 的进位信号为最终的进位信号。

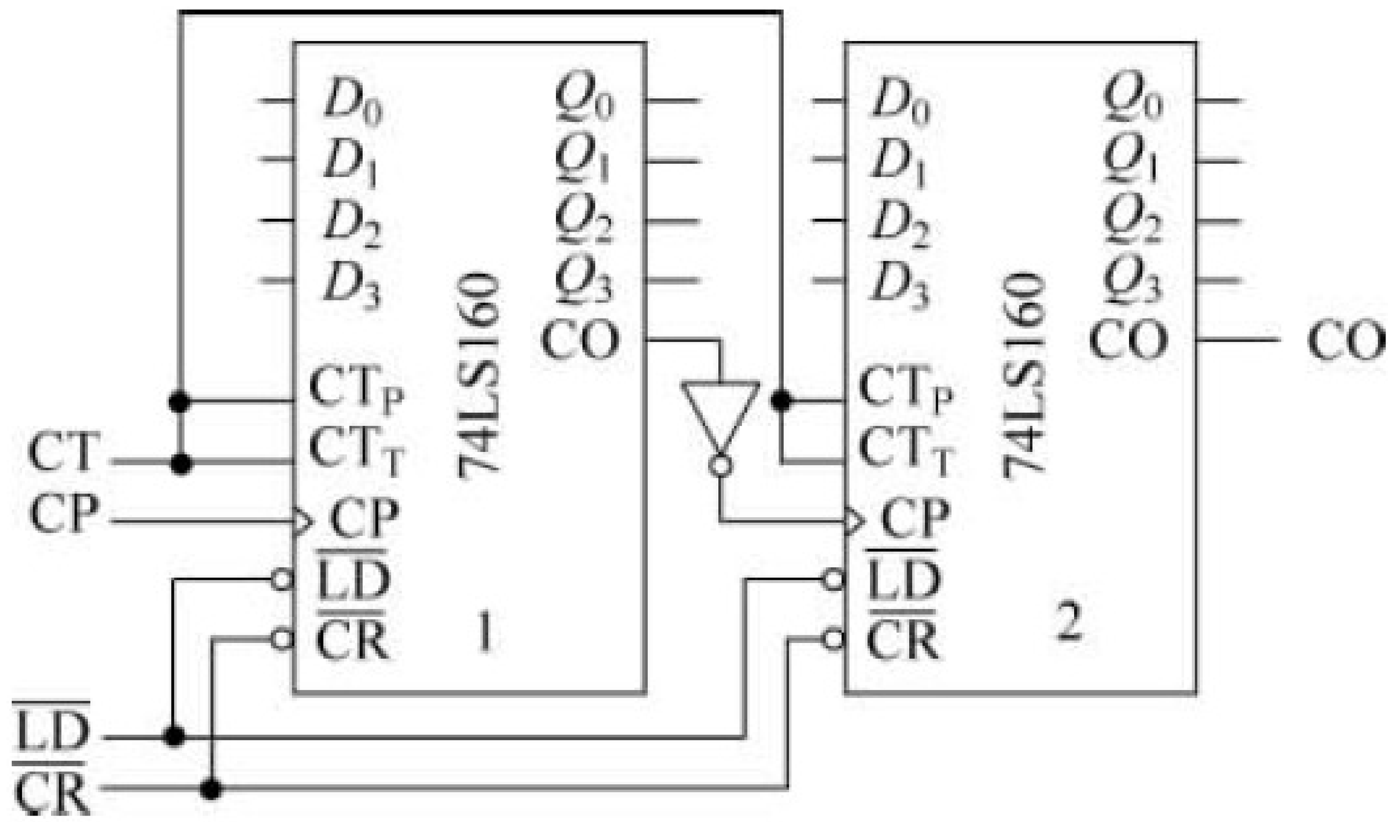


图 6.4.58 串行级联实例的图 1

显然，如图6.4.58所示电路正常计数时，片1从0～9循环计数，片2也从0～9循环计数。片2的计数信号为片1的溢出信号，因此，片1溢出时片2计数1次。片1、片2整体构成一个百进制计数器。片1的输出对应十进制数的个位，片2的输出对应十进制数的十位。

因为片1溢出信号为下降沿，取反后的溢出信号尽管满足要求，但将导致片2的状态变化和片1相比存在超过一个反相器传输时间的延时，可见，如图6.4.58所示电路为异步时序逻辑电路，性能方面弱于同步时序电路。

利用芯片的级联端进行级联的实例如图6.4.59所示。图中，片1、片2均为上升沿计数器。溢出时，74LS191的级联信号取自时钟信号，为上升沿。片1的级联信号接片2的时钟信号。两个芯片的计数允许、置数、加/减控制等控制端相互短接，片2的进位/借位信号、级联信号为最终的进位/借位信号、级联信号。

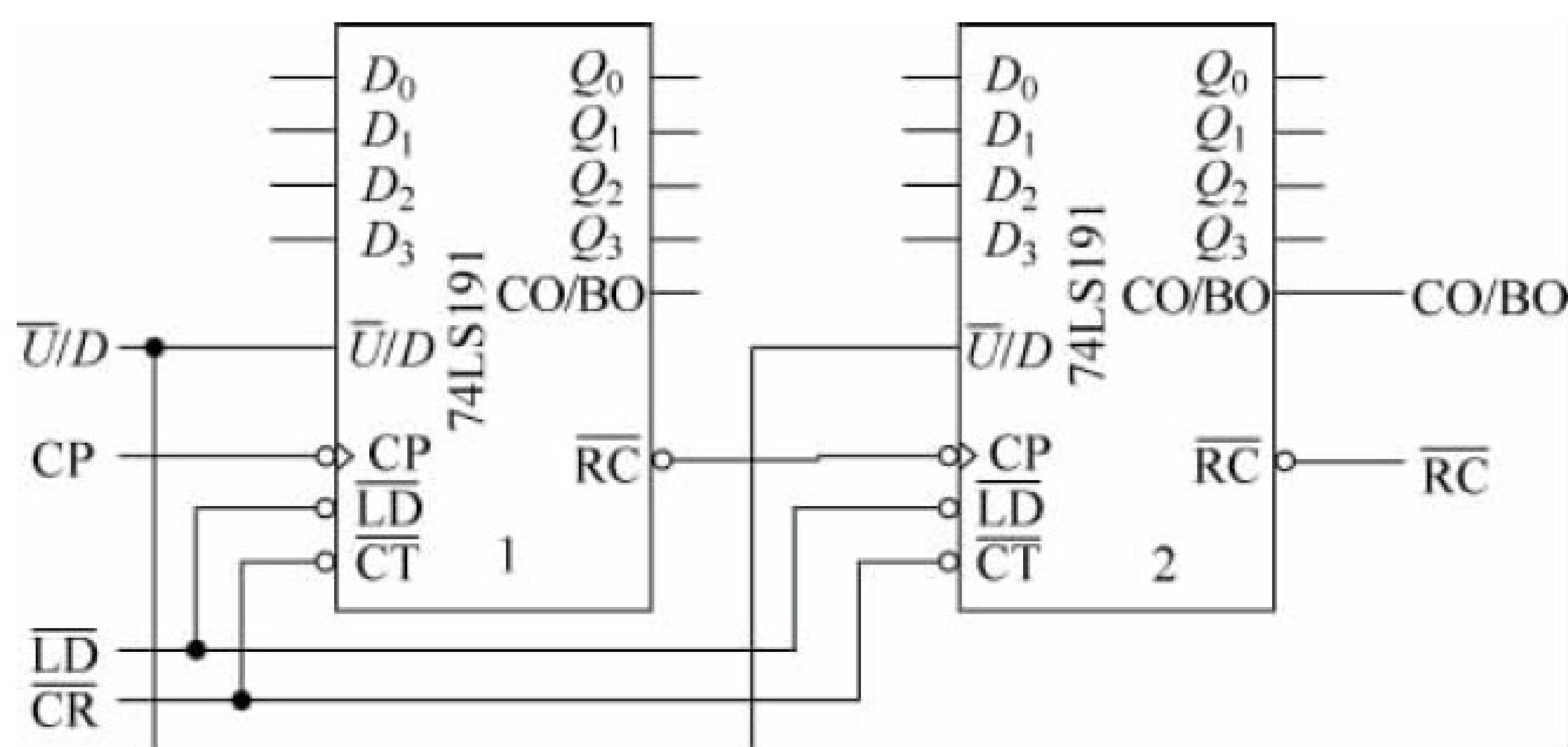


图6.4.59　串行级联实例的图2

显然，如图6.4.59所示电路正常计数时，片1从0～F循环计数，片2也从0～F循环计数。片2的计数信号为片1的溢出信号，因此，片1溢出时片2计数1次。片1、片2整体构成一个二百五十六进制计数器。片1的输出对应低4位，片2的输出对应高4位。

对照图6.4.58和图6.4.59，尽管如图6.4.59所示电路的时钟信号为片1的级联信号，但该信号取自时钟信号，电路的性能及可靠性等有保障，因此，对具有级联端的计数芯片，采用串行级联方法十分方便。

3. 并行级联方法

并行级联实例如图6.4.60所示。图中，片1、片2均为上升沿计数器。74LS160为满幅输出上升沿、溢出输出下降沿的计数器，片2的计数允许接片1的进位信号。两个芯片的时钟输入端、置数、清零等控制端相互短接，片2的进位信号为最终的进位信号。

显然，如图6.4.60所示电路正常计数时，片1从0～9循环计数，片2也从0～9循环计数。片2的计数允许为片1的溢出信号，因此，当片1满幅时，片2的计数允许有效。即片1进入状态“9”时，进位信号为1，允许片2计数，下个脉冲上升沿到来时，片1溢出，片2加1，片1、片2整体构成一个百进制计数器。片1的输出对应十进制数的个位，片2的输出对应十进制数的十位。

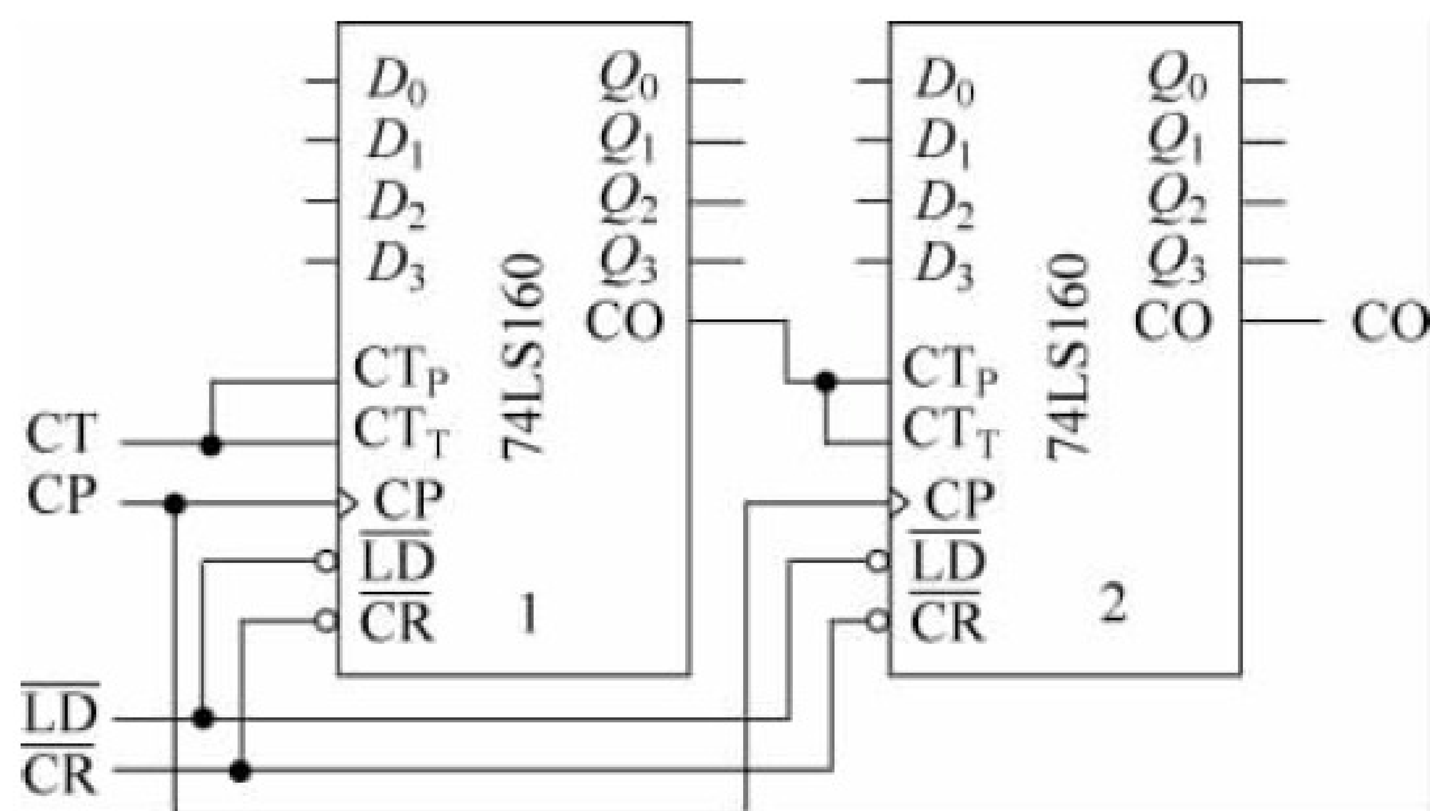


图 6.4.60 并行级联实例的图

片 1、片 2 的时钟输入端均来自外部时钟，因此如图 6.4.60 所示电路为同步时序电路。

【例 6.4.8】 试利用两片 74LS161 构成一个二十进制计数器。

解 可先用 74LS161 构成一个二百五十六进制计数器，之后，选用异步清零的方法实现二十进制计数器。

(1) 参考图 6.4.60，用两片 74LS161 构成一个二百五十六进制计数器

(2) 写出计数器 $Q_7 \sim Q_0$ 状态 S_N 的二进制代码，即

$$S_N = S_{20} = 10100$$

(3) 求出清零函数 $\overline{CR}$，有

$$\overline{CR} = \overline{Q_4 Q_2} \tag{6.4.11}$$

(4) 画出电路，如图 6.4.61 所示。

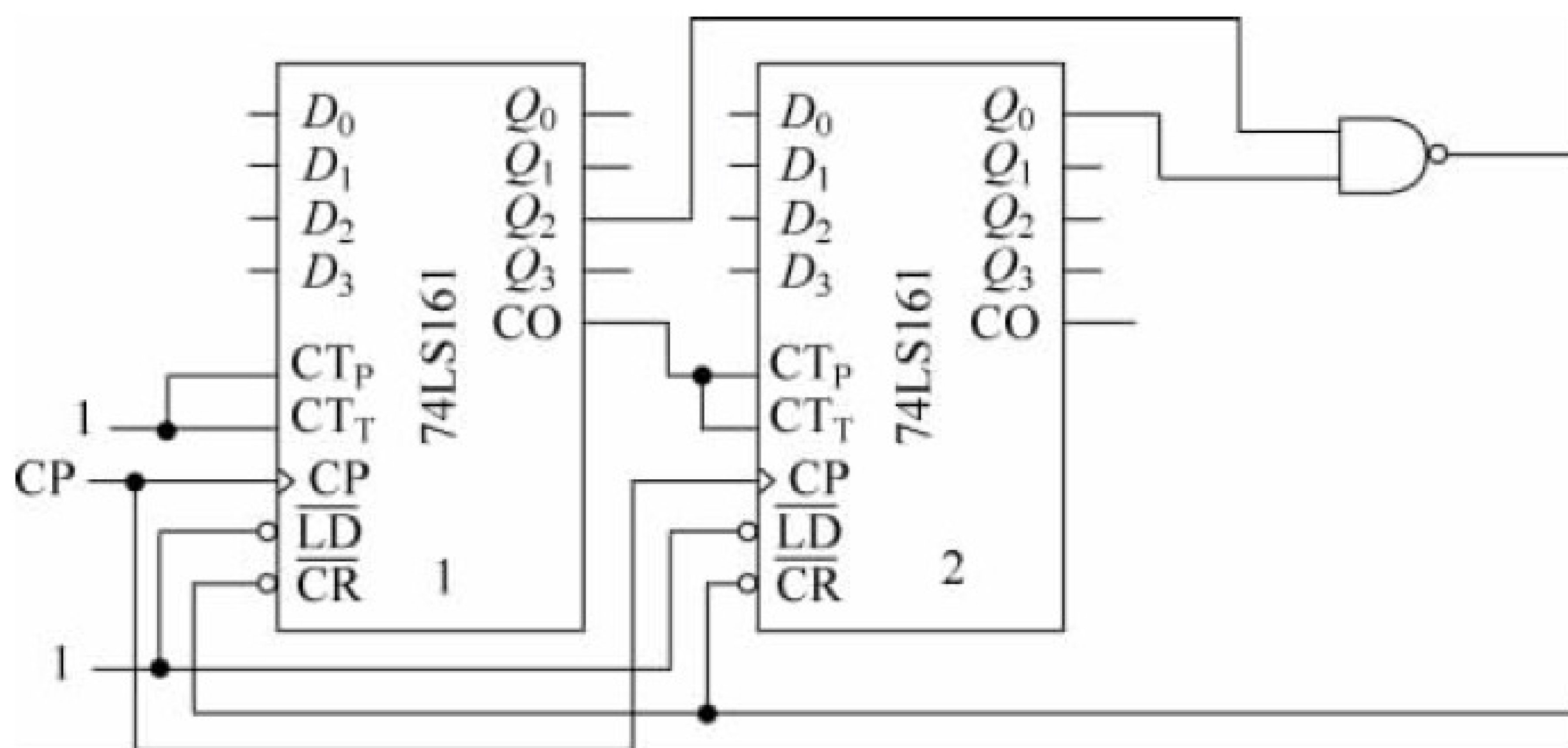


图 6.4.61 例 6.4.8 的图 1

当然，也可用两片 74LS160 构成一个二十进制计数器，参考电路如图 6.4.62 所示。

图 6.4.62 中，片 1、片 2 均对时钟信号计数，整体构成一个同步百进制加法计数器。当 $Q_5Q_4Q_3Q_2Q_1Q_0 = 10_{10}\ 0000_{10} = 20_{10}$ 时，计数器整体异步清零，$N=20$，为二十进制计数器。

也可按照级联的含义分析如图 6.4.62 所示电路。图中，片 1 计数允许一直有效，为一个十进制计数器。片 2 接清零控制电路，当 $Q_3Q_2Q_1Q_0 = 0010_2$ 时，计数器异步清零，$N=2$，为二进制计数器。片 2 的计数允许接片 1 的进位信号，即当片 1 溢出时片 2 加 1，因此，当片 2 进入状态“2”时，片 1 的状态为“0”，片 2 的清零信号对片 1 没有影响。

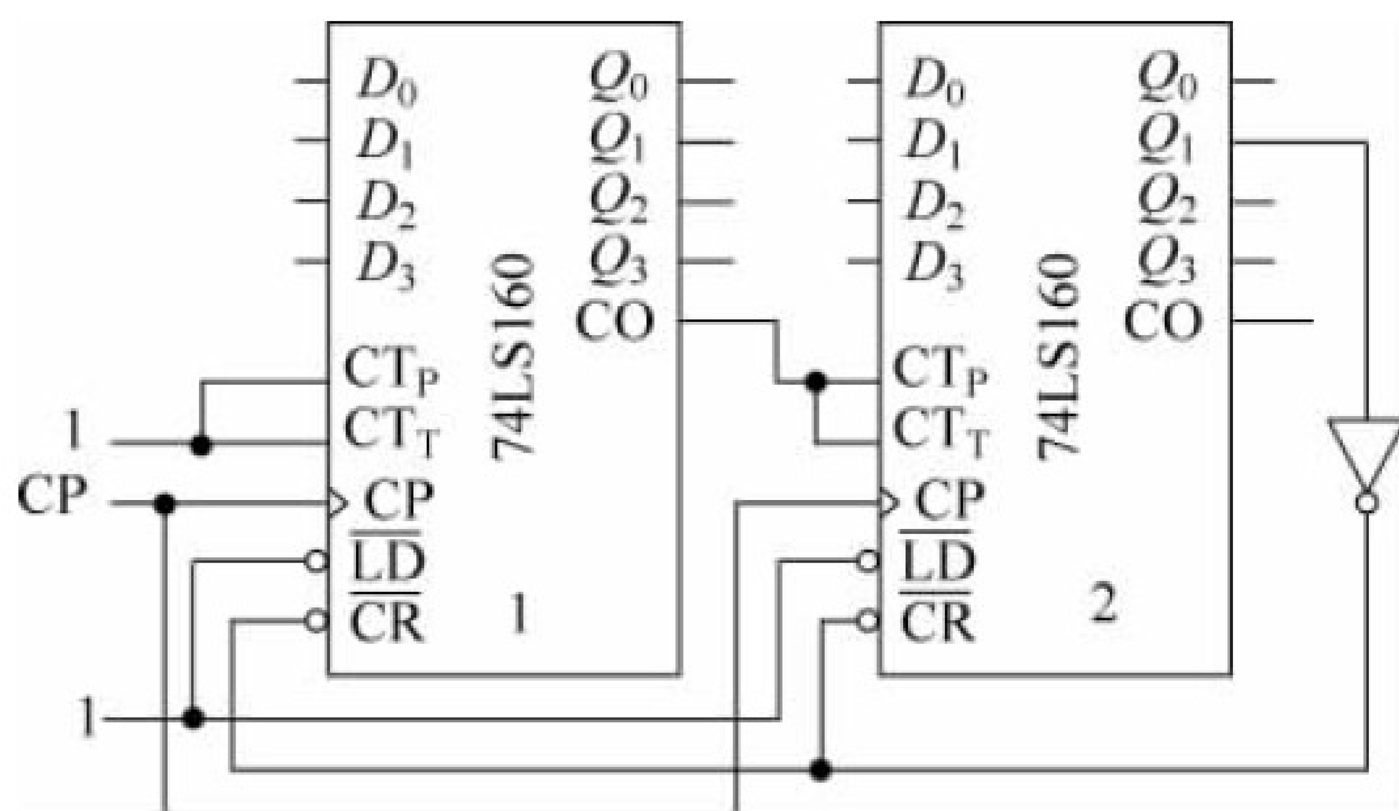


图 6.4.62 例 6.4.8 的图 2

因此，如图 6.4.62 所示电路的计数容量＝片 1×片 2＝20，为二十进制计数器。

【**例 6.4.9**】 试分析如图 6.4.63 所示电路的逻辑功能。

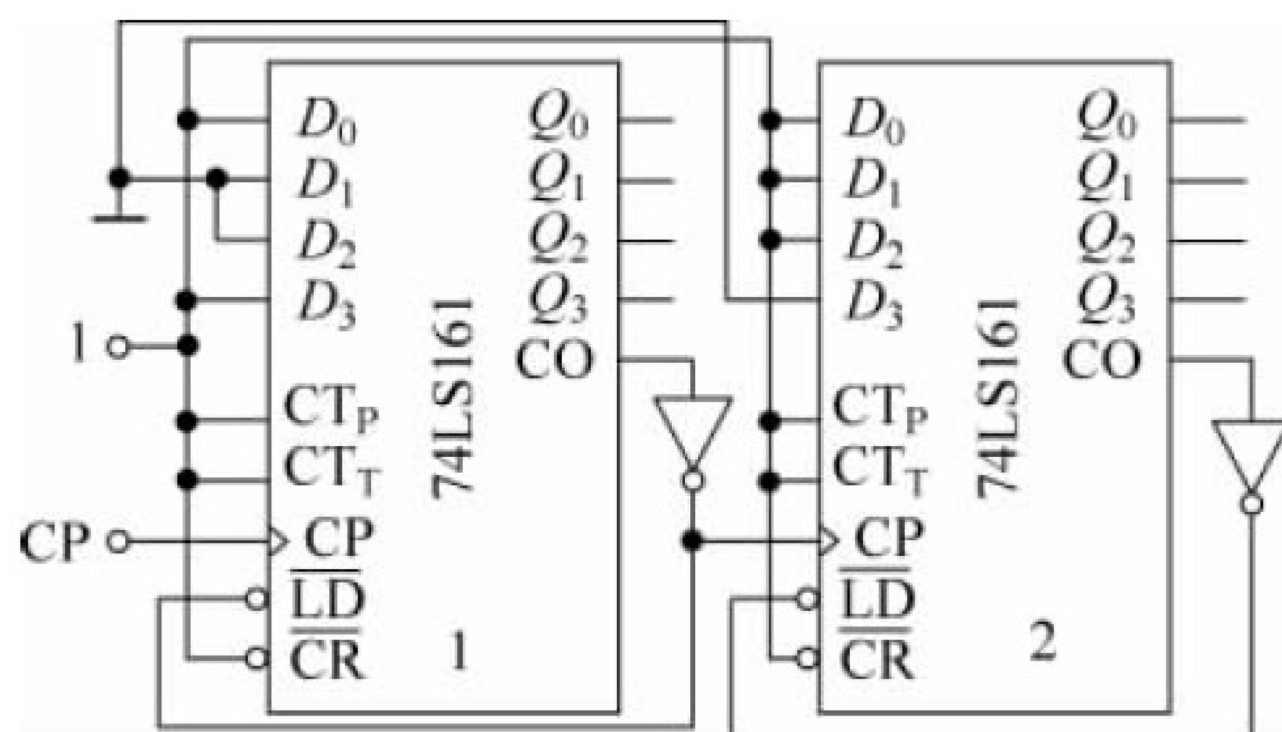


图 6.4.63 例 6.4.9 的图 1

解 （1）接法分析。如图 6.4.63 所示电路由两片 74LS161 和两个非门组成。两片 74LS161 的$\overline{CR}$、CT_P、CT_T均接高电平，$\overline{LD}=\overline{CO}$。片 1 的 $D_3D_2D_1D_0=1001$，片 2 的 $D_3D_2D_1D_0=0111$。可见，当$\overline{LD}$无效时，计数器处于正常计数状态。当计数器计数计到最大值时，$\overline{CO}=0$。当下一个计数脉冲上升沿到来时，计数器置数，进入 $D_3D_2D_1D_0$设置的状态。

（2）功能分析。片 1 的仿真结果如图 6.4.64 所示。从仿真结果可看出，片 1 为七进制加法计数器（预置数为 9，进入 15 时置数信号有效，共 7 个有效状态）。片 2 的仿真结果如图 6.4.65（图中，$\overline{CO_1}$、$\overline{CO_2}$ 为片 1、片 2 进位输出信号取反）。从仿真结果可看出，芯片 2 为九进制加法计数器（预置数为 7，进入 15 时置数信号有效，共 9 个有效状态）。

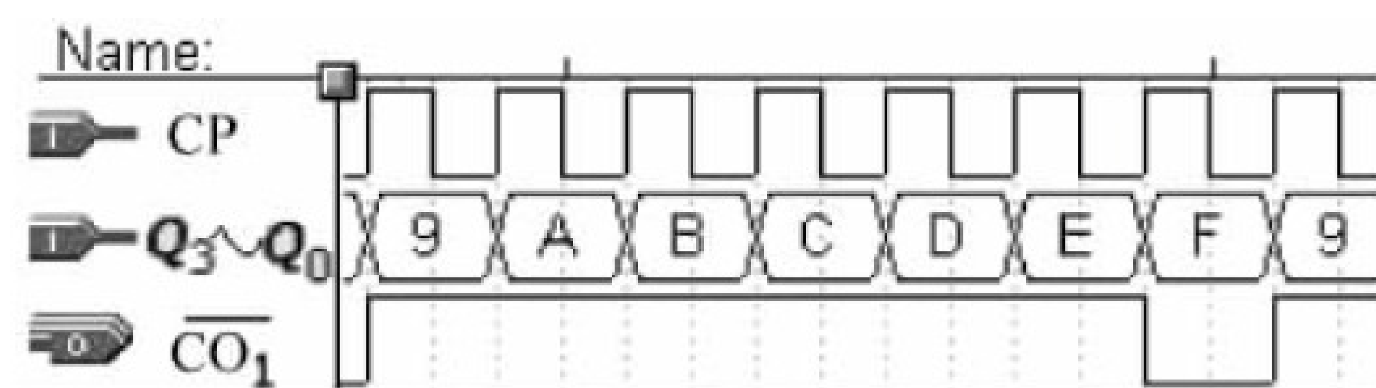


图 6.4.64 例 6.4.9 的图 2

片 2 的计数脉冲为片 1 的进位脉冲。而片 1 每计 7 个 CP 计数脉冲产生一个进位输出信号，7×9＝63，所以，如图 6.4.63 所示电路为六十三进制计数器。

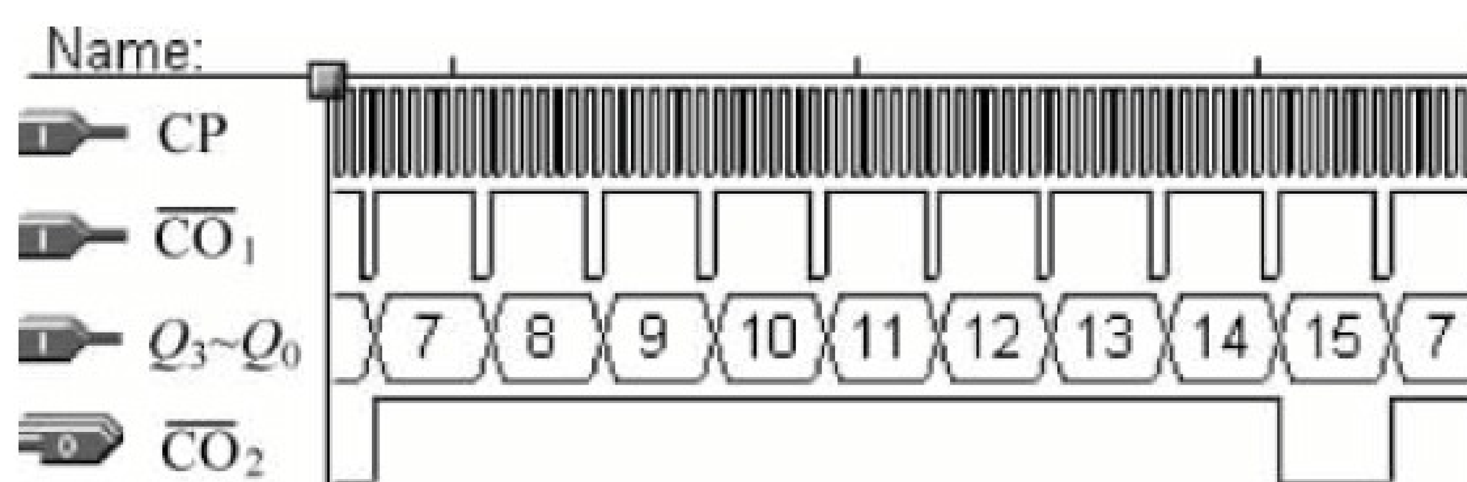


图 6.4.65 例 6.4.9 的图 3

6.4.8 移位寄存器型计数器

也可用移位寄存器构成计数器,称为移位寄存器型计数器,简称移存型计数器。常用的移存型计数器有环形计数器和扭环形计数器。



1. 环形计数器

【例 6.4.10】 试分析如图 6.4.66 所示电路的逻辑功能。

解 (1) 接法分析。如图 6.4.66 所示电路为用 4 位移位寄存器 74LS195 组成的环形计数器,74LS195 功能表如表 6.3.1 所示。$SH/\overline{LD}$移位置数控制端的负脉冲为启动信号,$J=\overline{K}=Q_3$,预置数 $d_0d_1d_2d_3=1000$。

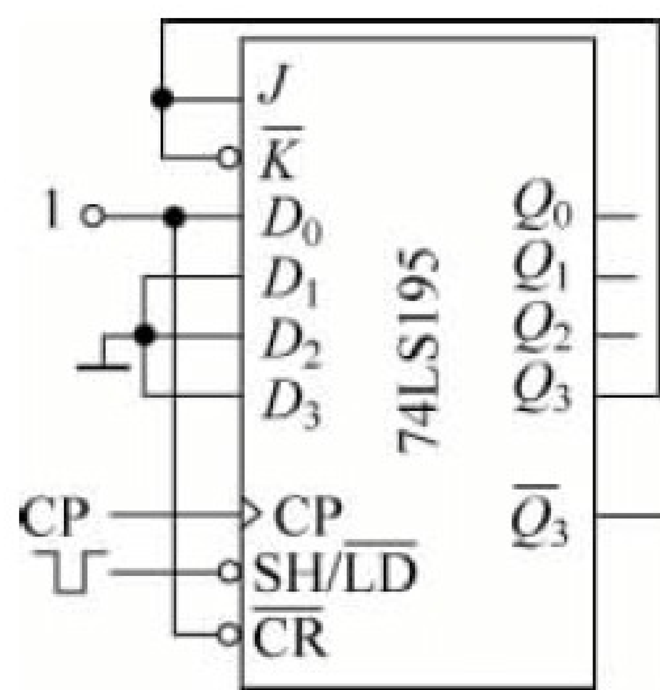


图 6.4.66 例 6.4.10 图 1

(2) 工作原理。

① 并行送数。在启动负脉冲作用下,当 $SH/\overline{LD}=0$ 时,由 74LS195 功能表的第 2 行可知,在 CP 脉冲的作用下将并行置入的数据 $d_0d_1d_2d_3=1000$ 送入移位寄存器中。

② 循环右移串行送数操作。当$\overline{CR}=1$,$SH/\overline{LD}=1$ 时,移位寄存器正常工作,执行右移移位操作,有

$$Q_i = Q_{i-1} \tag{6.4.12}$$

即

$$Q_1 = Q_0,\quad Q_2 = Q_1,\quad Q_3 = Q_2$$

$$Q_0 = J\overline{Q}_0^n + \overline{K}Q_0^n$$

因 $J=\overline{K}=Q_3$,所以有

$$Q_0 = Q_3 \tag{6.4.13}$$

可见,正常工作时,如图 6.4.66 所示电路在 CP 脉冲的作用下,将寄存器中预置数据 0001($Q_3Q_2Q_1Q_0$)循环右移,可求得 $Q_3Q_2Q_1Q_0$状态循环如下:

$$0001 \rightarrow 0010 \rightarrow 0100 \rightarrow 1000 \rightarrow 0001 \rightarrow \cdots$$

由式(6.4.12)和式(6.4.13),可总结环形计数器的数据移动特点为首尾串接循环移动。

(3) 计算机仿真。如图 6.4.66 所示电路的计算机仿真结果如图 6.4.67 所示。在启动

信号作用下，移位寄存器存入数据 0001($Q_3Q_2Q_1Q_0$)，然后一直进行右移操作，实现了模值为 4 的计数功能。由于这种移存型计数器，在每一个输出端轮流出现 1(或者 0)，故称为环形计数器。

(4) 自启动改进。显然，当预置数为“0000”“0111”“0011”“1111”等诸多状态时，循环移位均会构成另一个无效循环，如图 6.4.66 所示电路不能自启动，改进电路如图 6.4.68 所示。图中，右移移位控制一直有效，有兴趣的读者可自行分析该电路自启动的实现方法。

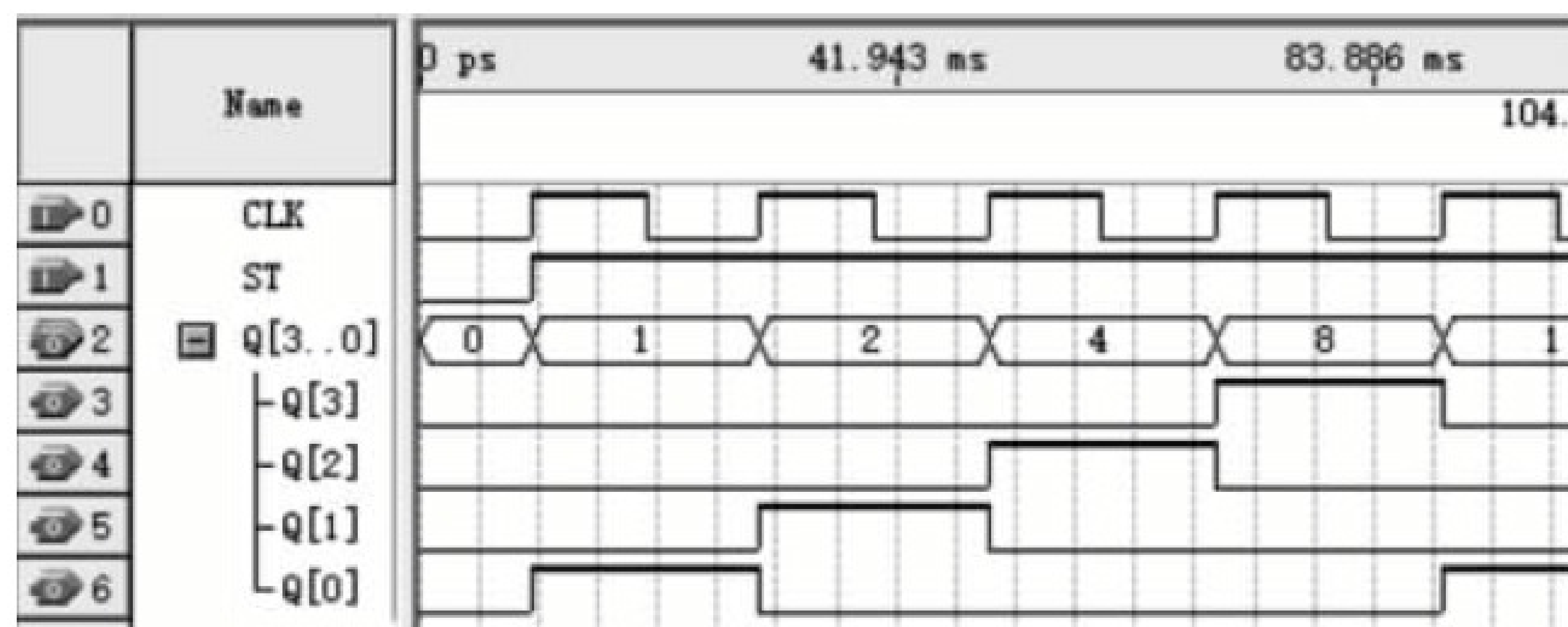


图 6.4.67 例 6.4.10 图 2

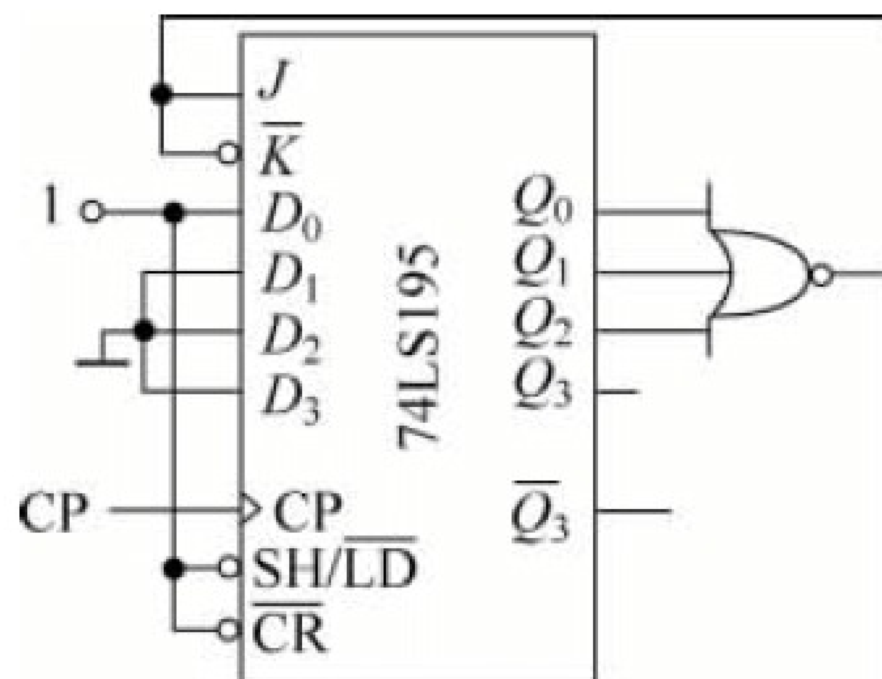


图 6.4.68 例 6.4.10 图 3

2. 扭环形计数器



【例 6.4.11】 试分析如图 6.4.69 所示电路的逻辑功能。

解 (1) 接法分析。如图 6.4.69 所示电路为用 4 位移位寄存器 74LS195 组成的扭环形计数器。$SH/\overline{LD}$移位置数控制端的负脉冲为启动信号，预置数 $d_0d_1d_2d_3=0000$。对照图 6.4.66，二者接法的主要区别是本例中 J、$\overline{K}$ 端接至 $\overline{Q}_3$ 端，图 6.4.66 中 $J=\overline{K}=Q_3$。

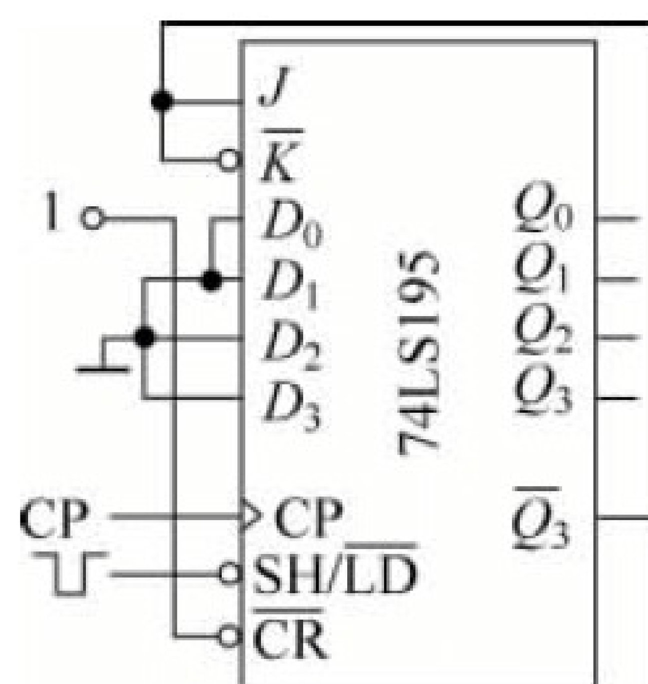


图 6.4.69 例 6.4.11 图 1

(2) 工作原理。

① 并行送数。在启动负脉冲作用下，当 $SH/\overline{LD}=0$ 时，由 74LS195 功能表的第 2 行可知，在 CP 脉冲的作用下将并行置入的数据 $d_0d_1d_2d_3=0000$ 送入移位寄存器中。

② 循环右移串行送数操作。当$\overline{CR}=1$，$SH/\overline{LD}=1$ 时，移位寄存器正常工作，执行右移移位操作，$Q_i=Q_{i-1}$。

$$Q_0 = J\overline{Q}_0^n + \overline{K}Q_0^n$$

因 $J=\overline{K}=\overline{Q}_3$，所以有

$$Q_0 = \overline{Q}_3 \tag{6.4.14}$$

可见，正常工作时，如图 6.4.69 所示电路在 CP 脉冲的作用下，将预置数据 0000($Q_3Q_2Q_1Q_0$)右移，但 $Q_0=\overline{Q}_3$，可求得 $Q_3Q_2Q_1Q_0$状态循环如下：

$$0000 \rightarrow 0001 \rightarrow 0011 \rightarrow 0111 \rightarrow 1111 \rightarrow 1110 \rightarrow 1100 \rightarrow 1000 \rightarrow 0000$$

由式(6.4.12)和式(6.4.14),可总结扭环形计数器的数据移动特点为首接尾反循环移动。状态变化规律为由全0变化到全1,之后又变化到全0,这种特点的计数器称为扭环形计数器(也称为约翰逊计数器)。

(3) 计算机仿真。如图6.4.69所示电路的计算机仿真结果如图6.4.70所示。由仿真结果可看出,在启动信号作用下,移位寄存器存入数据0000($Q_3Q_2Q_1Q_0$),然后一直进行右移操作($Q_0=\overline{Q}_3$),输出端1的个数逐渐增加到最大,之后,逐渐减少到0,实现了模值为8的计数器。

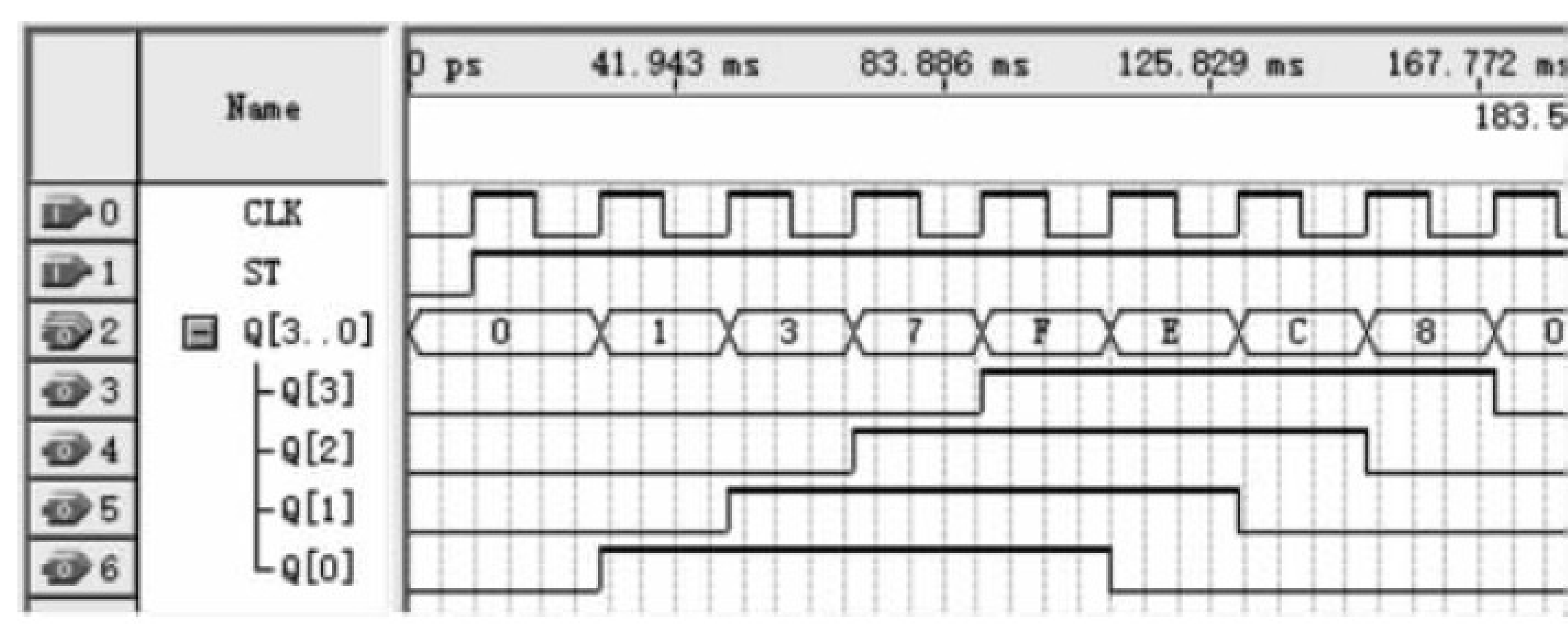


图6.4.70 例6.4.11图2

(4) 自启动改进。如图6.4.69所示电路不能自启动,改进电路如图6.4.71所示。图中,右移移位控制一直有效。有兴趣的读者可自行分析该电路自启动的实现方法。

也可用移位寄存器实现其他模值的计数器,如图6.4.72所示电路的计算机仿真结果如图6.4.73所示。可令初值为"0000",根据74LS195的逻辑功能可得出图示仿真结果。从仿真结果表可看出,如图6.4.72所示电路为由移位寄存器构成的模值为13的计数器。

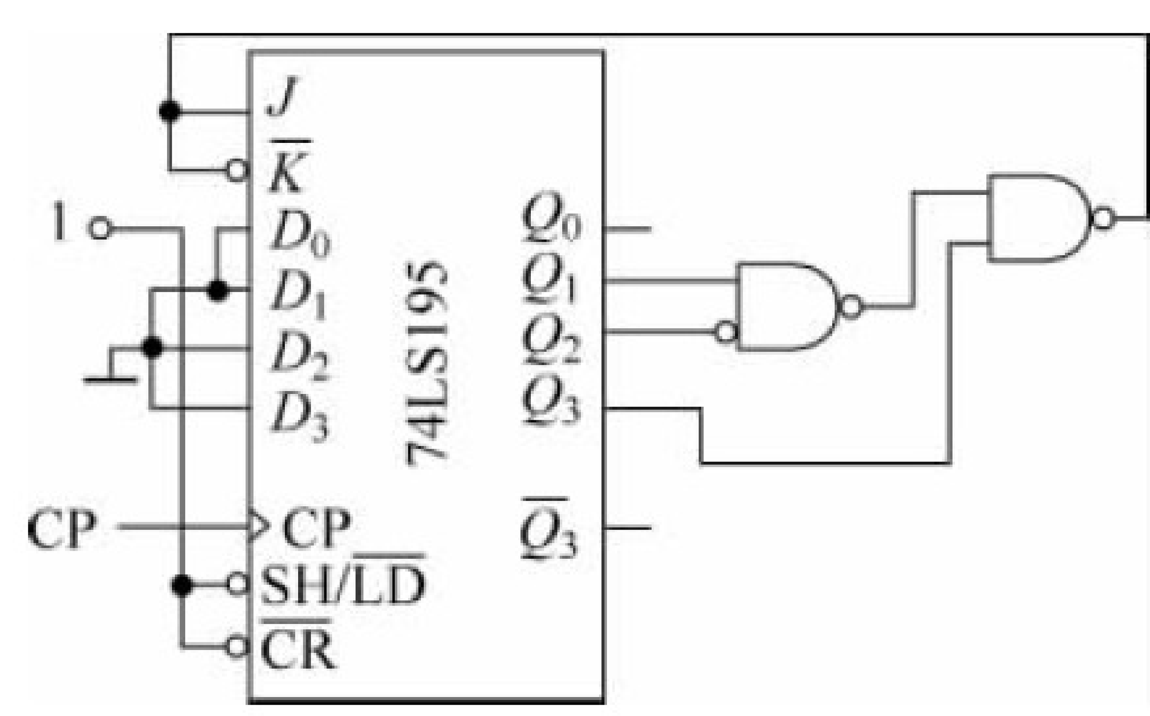


图6.4.71 例6.4.11图3

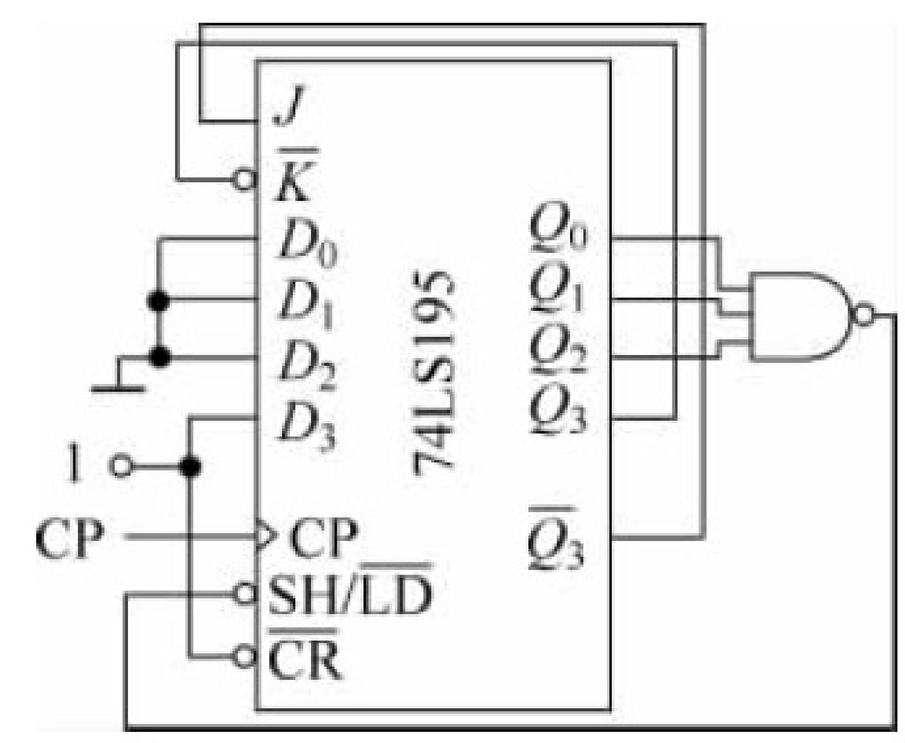


图6.4.72 十三进制计数器

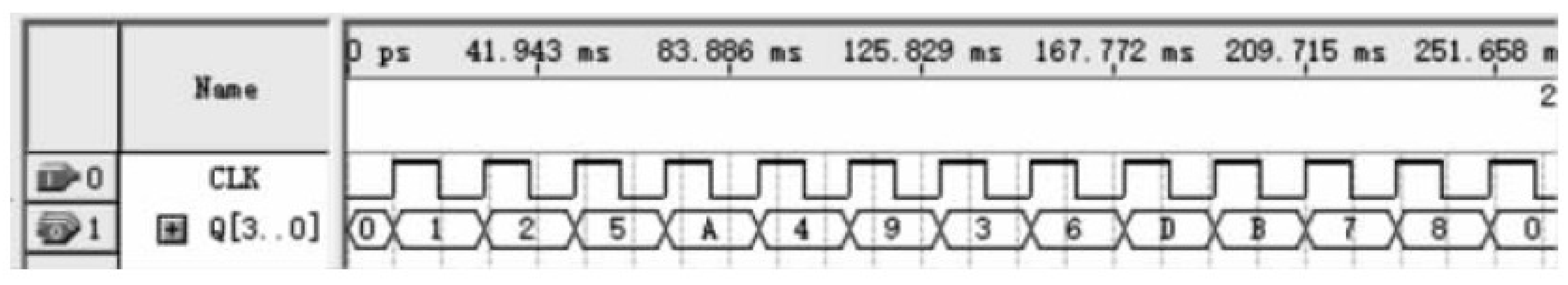


图6.4.73 图6.4.72所示电路的计算机仿真结果

如果在如图6.4.72所示电路中改变并行输入的数据$d_0 \sim d_3$,还可以获得其他模值数的计数器。

3. 移存型计数器的应用

(1) 构成顺序脉冲发生器。在数字电路中,能按一定时间、一定顺序轮流输出脉冲波形的电路称为顺序脉冲发生器。常用顺序脉冲发生器来控制某些设备按照事先规定的顺序进行运算或操作。

顺序脉冲发生器也称脉冲分配器或节拍脉冲发生器,可用移存型计数器构成,也可用计数器和译码器组合构成。

如图 6.4.68 所示环形计数器能够自启动,实现了在每一个输出端轮流出现 1,同时也构成了一个 4 路输出的顺序正脉冲发生器。将如图 6.4.68 所示环形计数器的输出 $Q_3Q_2Q_1Q_0$ 取反,则实现了在每一个输出端轮流出现 0,构成了一个 4 路输出的顺序负脉冲发生器。

参照如图 6.4.68 所示环形计数器的实现思路,可设计如图 6.4.74 所示的 3 路输出顺序脉冲发生器。图中,$Q_2Q_1Q_0$ 为 3 路输出,仿真波形如图 6.4.75 所示。从仿真结果表可看出,如图 6.4.74 所示电路构成了一个 3 路输出的顺序正脉冲发生器。

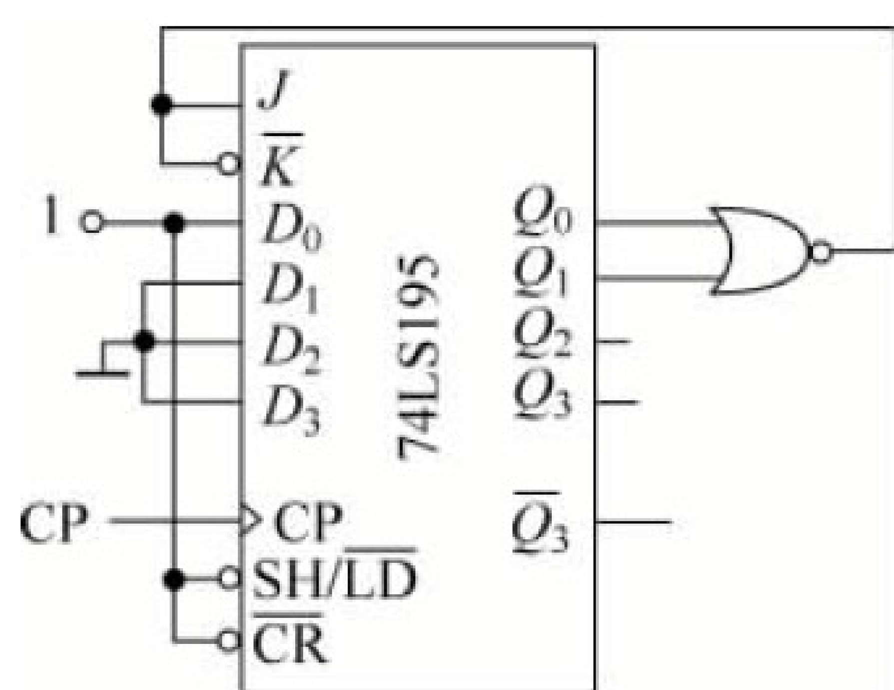


图 6.4.74　3 路输出顺序脉冲发生器

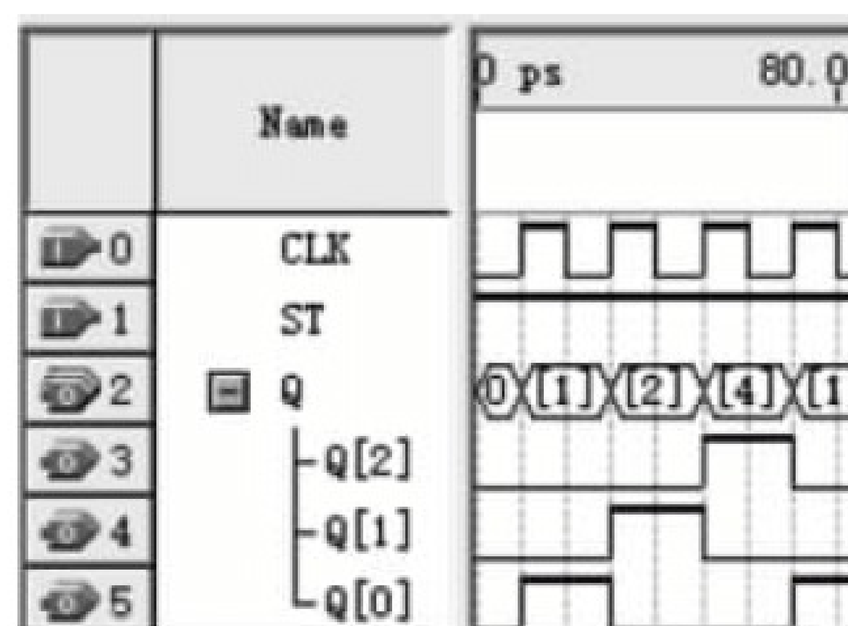


图 6.4.75　仿真波形

(2) 构成序列信号发生器。序列信号是指在同步脉冲作用下循环地产生一串周期性的二进制信号,能产生这种信号的逻辑器件称为序列信号发生器或序列发生器。

移位寄存器可存储一串二进制信号,利用循环移位可方便地周期性地产生存储的这串二进制信号,常用于构成序列信号发生器。有兴趣的读者可参考相关书籍。也可用计数器结合数据选择器实现。

复习与思考

6.4.1　分析如图 6.4.76 所示电路的逻辑功能,并说明它与图 6.4.1 所示电路相比有何优点。

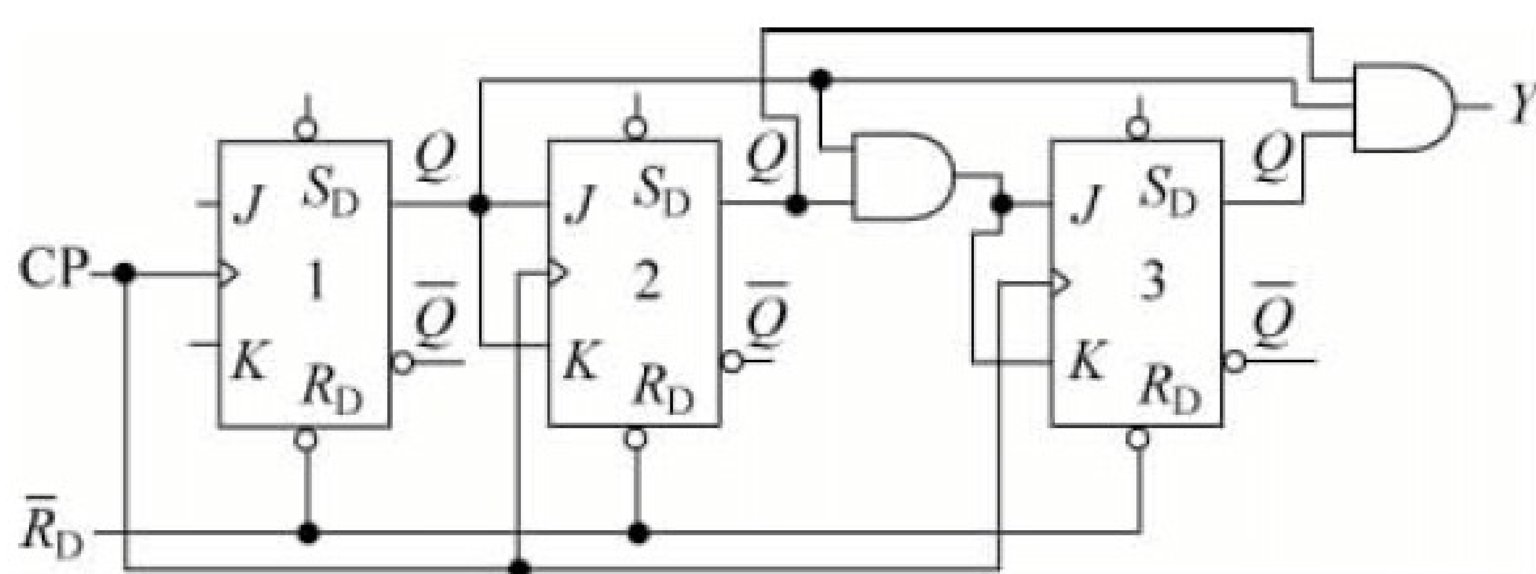


图 6.4.76　复习与思考 6.4.1 的图

6.4.2　给你一片 74LS161 和一个主从 JK 触发器，可构成模值最大为多少的计数器？

6.4.3　试结合如图 6.4.14 所示 74LS161 原理图总结计数允许 CT_T、CT_P的区别。

6.4.4　试总结 74LS191 的 CO/BO 信号及级联信号的特点。

6.4.5　为什么 74LS161 的计数允许为高电平有效，74LS191 的计数允许（使能端）却是低电平有效？

6.4.6　直接利用 JK 触发器功能特点求出如图 6.4.30 所示电路的仿真时序图。

6.4.7　分析图 6.4.77 所示电路为多少进制计数器。

6.4.8　分析图 6.4.78 所示电路为多少进制计数器。

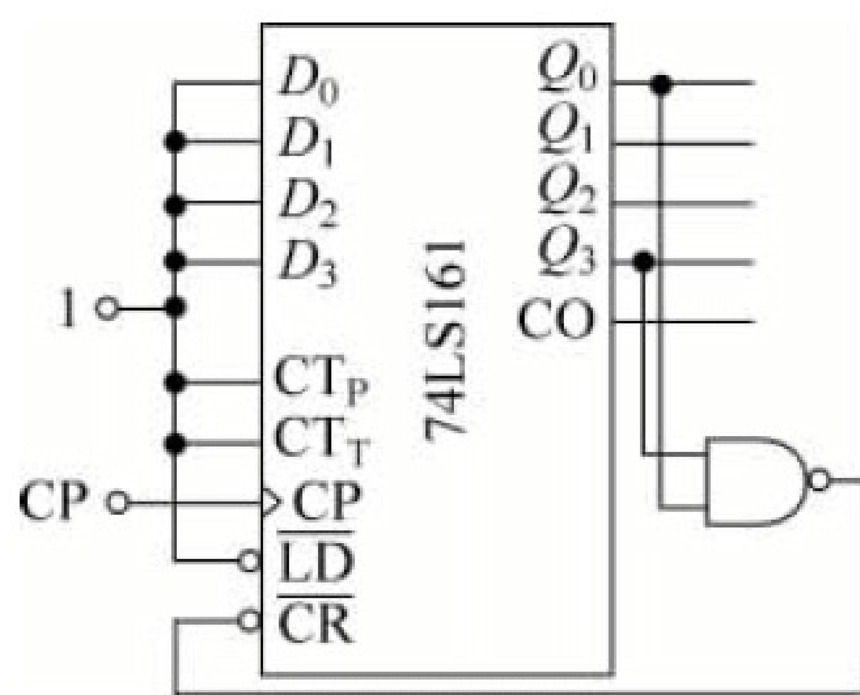


图 6.4.77　复习与思考 6.4.7 的图

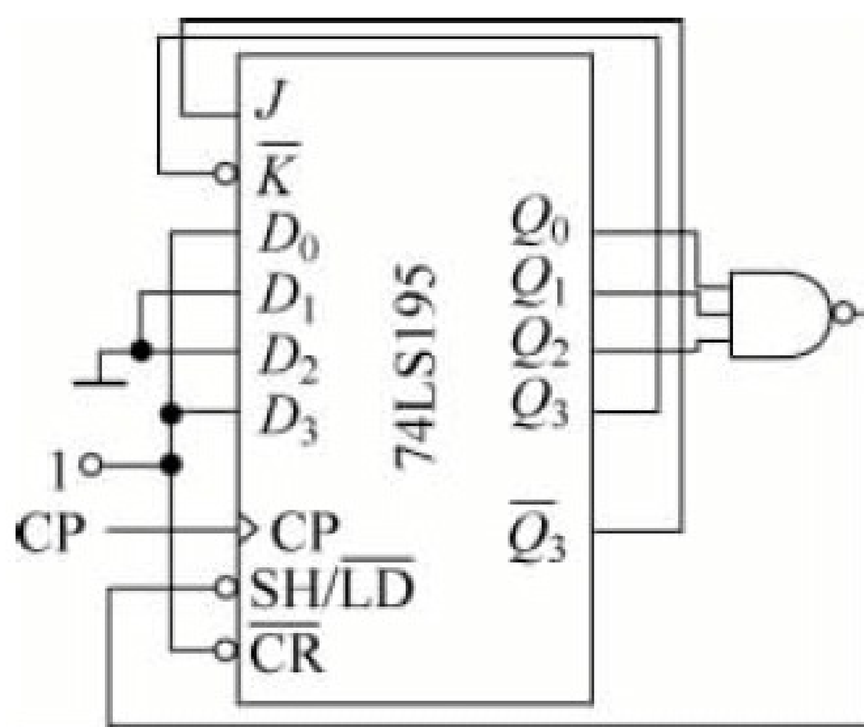


图 6.4.78　复习与思考 6.4.8 的图

6.5　时序逻辑电路的设计

设计时序逻辑电路就是要求设计者根据具体的逻辑问题，设计出实现这一逻辑功能要求的时序逻辑电路。设计的主要方法有利用触发器、门电路等小规模器件实现时序逻辑电路，利用计数器、寄存器、译码器等中规模器件结合门电路实现时序逻辑电路及基于超大规模集成电路采用硬件描述语言描述实现时序逻辑电路等。

*6.5.1　用小规模集成电路设计同步时序逻辑电路

用触发器、门电路等小规模器件设计同步时序逻辑电路要求电路最简。所设计的电路最简的标准：使用的触发器和门电路的数目最少，而且触发器和门电路的输入端数目也最少。

用小规模集成电路设计同步时序电路一般步骤如下。

(1) 分析设计要求，建立原始状态图或原始状态表。分析设计所提出的逻辑功能要求，设置输入变量、输出变量，以及电路的状态数和电路内部状态的关系，这项工作称为状态设置。

根据电路内部状态间的关系画出原始状态图或者原始状态表。

(2) 进行状态化简，求最简状态图。若有两个电路状态，在输入输出相同的条件下转换到的新状态也相同，这样的两个状态称为**等价状态**。等价状态可以合并，合并后的状态称

为最简状态图。

显然，电路中的状态数越少，设计出来的电路也就越简单。

(3) 到状态分配，画出用二进制码进行编码后的状态图。

① 确定触发器的数目 n。时序电路的状态是用触发器状态的不同组合来表示的。因为 n 个触发器共有 2^n 种状态组合，所以对需要 r 个状态的时序电路，可由下式求出所需的触发器数目 n：

$$2^{n-1} < r < 2^n \tag{6.5.1}$$

② 状态分配。状态分配又称为状态编码。对于 n 位二进制代码有 2^n 种不同取值，可用来对 r 个状态进行分配，其方案很多。如果选择恰当，则可得到比较简单的设计结果。反之，如果方案选得不好，设计出来的电路往往会很复杂，甚至还不能自启动，所以，状态分配既有技巧问题，也与经验有关。

③ 画出编码后的状态图。

(4) 选取触发器，求出驱动方程，输出方程。由于 JK 触发器的逻辑功能齐全使用灵活，D 触发器控制简单设计容易，所以一般设计都选用 JK 触发器或 D 触发器。

选取触发器，根据状态分配后的状态图写出状态转换真值表。然后由状态转换真值表经卡诺图求出激励(驱动)函数，输出函数方程。

(5) 检查所设计出的电路能否自启动。如果在状态分配中出现无效状态，则应先检查能否自启动。检查方法是根据所求的驱动方程代入该触发器的特性方程，得到状态方程。然后设电路的原状态分别为各无效状态，当 CP 时钟脉冲到来时代入状态方程求出新状态。若能进入到有效状态，则电路能够自启动；否则重新进行分配，重复上述过程直至电路能够自启动为止。

(6) 根据所求出的驱动方程、输出方程画出电路图。

下面举例介绍同步时序逻辑电路的设计方法。

【例 6.5.1】 试设计一个五进制的同步计数器。

解 显然这是一个 Mealy 型的同步时序电路的设计。具体设计步骤如下。

(1) 分析逻辑功能要求，设置状态，建立原始状态图。显然，模值为 5 的计数器是指“逢五进一”的计数器，它的计数范围为 0～4，共 5 个状态。计数器有一个输入端和一个输出端，输入端用于输入计数脉冲，输出端 Y 输出进位信号。

令 S_0 为电路的初始状态，输入 1 个计数脉冲后进入到状态 S_1，输入 4 个计数脉冲后进入到状态 S_4，同时向高位输出一个进位信号($Y=1$)，当第 5 个计数脉冲到来后计数器返回到初始状态 S_0。根据逻辑功能要求，可画出如图 6.5.1 所示的原始状态图。

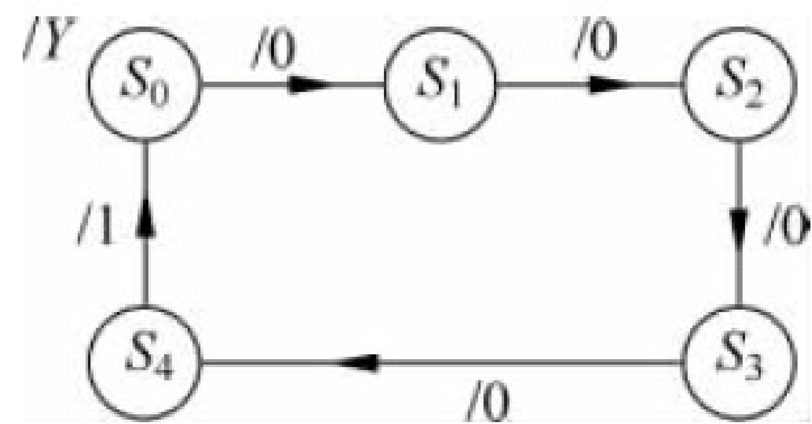


图 6.5.1　例 6.5.1 的图 1

(2) 状态分配。五进制计数器具有 5 个状态，原始状态图中只有 5 个状态，因此不存在等价状态，可直接进行状态分配。

原始状态图中共有 5 个状态，应选取 3 个触发器用于分配

状态，其状态分配 $S=Q_2Q_1Q_0$，令 $S_0=000$，$S_1=001$，$S_2=010$，$S_3=011$，$S_4=100$，可画出如图 6.5.2 所示的状态分配后的状态图。

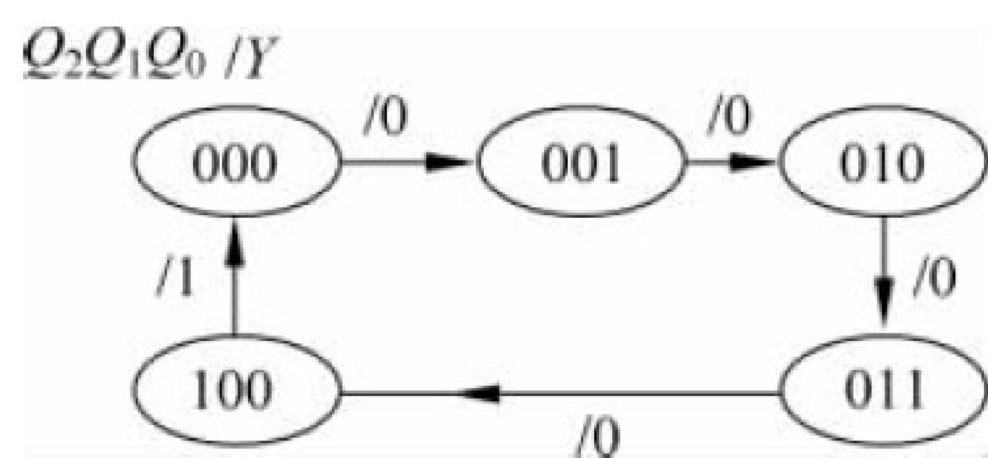


图 6.5.2　例 6.5.1 的图 2

(3) 确定触发器类型，求出触发器的驱动方程和电路的输出方程。

① 写出状态转换真值表(选用 JK 触发器)。以 Q_2^n、Q_1^n、Q_0^n作为自变量，Q_2^{n+1}、Q_1^{n+1}、Q_0^{n+1}、J_2、K_2、J_1、K_1、J_0、K_0、Y 为函数，可列出如表 6.5.1 所示的状态转换真值表。表中的第 1、2、6 列由如图 6.5.2 所示的状态图转换得到；表中的 J_2K_2、J_1K_1、J_0K_0列由如表 5.3.2 所示的 JK 触发器激励表写出。

表 6.5.1　状态转换真值表

$Q_2^n\ Q_1^n\ Q_0^n$	$Q_2\ Q_1\ Q_0$	$J_2\ \ K_2$	$J_1\ \ K_1$	$J_0\ \ K_0$	Y
0　0　0	0　0　1	0　×	0　×	1　×	0
0　0　1	0　1　0	0　×	1　×	×　1	0
0　1　0	0　1　1	0　×	×　0	1　×	0
0　1　1	1　0　0	1　×	×　1	×　1	0
1　0　0	0　0　0	×　1	0　×	0　×	1
1　0　1	×　×　×	×　×	×　×	×　×	×
1　1　0	×　×　×	×　×	×　×	×　×	×
1　1　1	×　×　×	×　×	×　×	×　×	×

② 求出驱动方程、输出方程。由表 6.5.1 分别经过如图 6.5.3 所示的卡诺图可求出驱动方程，具体如下：

由图 6.5.3(a)、图 6.5.3(b)可求出 $J_2=Q_1Q_0$，$K_2=1$；

由图 6.5.3(c)、图 6.5.3(d)可求出 $J_1=Q_0$，$K_1=Q_0$；

由图 6.5.3(e)、图 6.5.3(f)可求出 $J_0=\overline{Q}_2$，$K_0=1$。

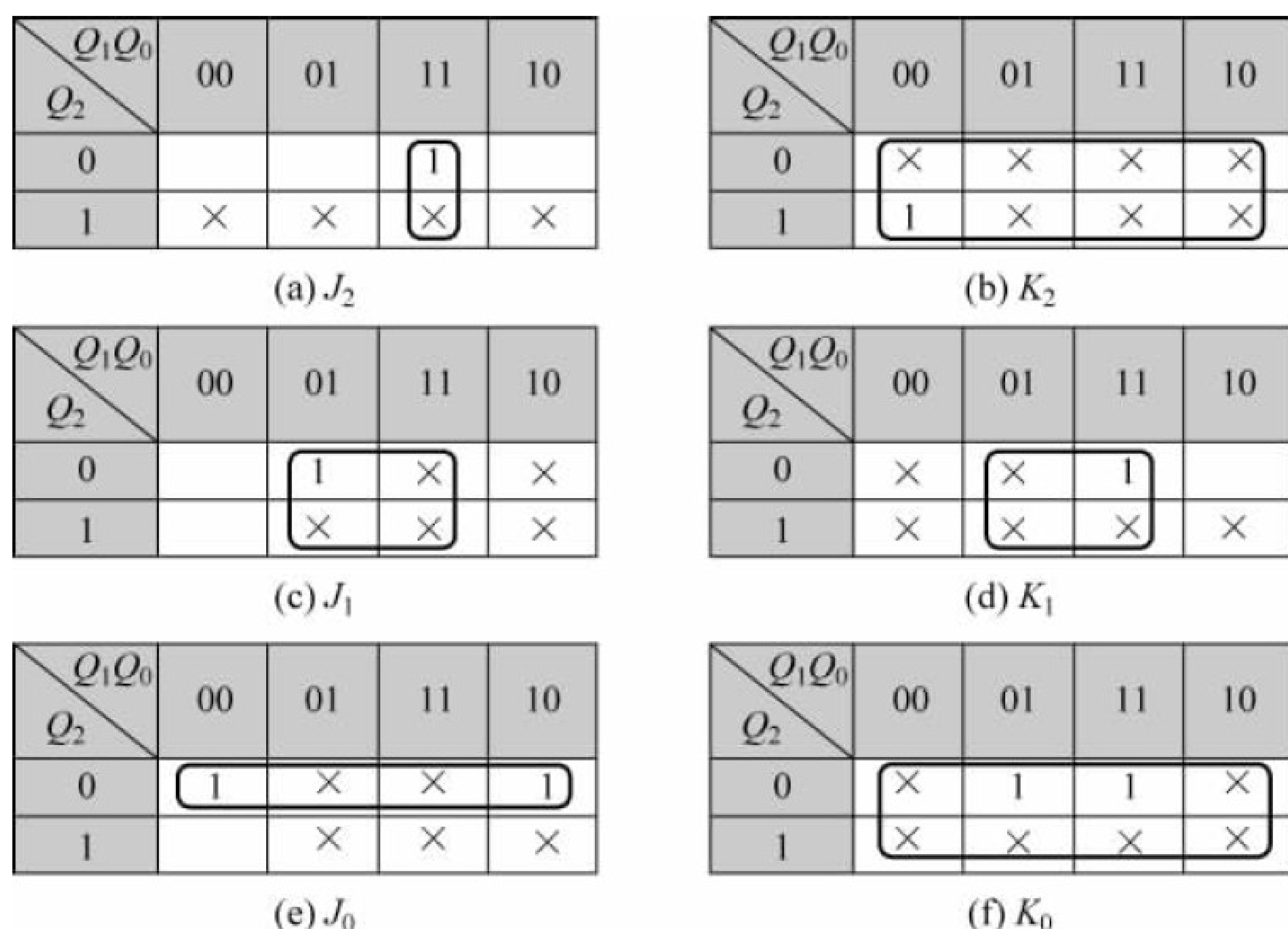


图 6.5.3　例 6.5.1 的图 3

经过如图 6.5.4 所示的卡诺图可求出输出方程，有

$$Y = Q_2$$

Q_2 \ Q_1Q_0	00	01	11	10
0				
1	1	×	×	×

图 6.5.4　例 6.5.1 的图 4

(4) 检查自启动。设计的电路存在无效状态，需检查所设计的电路能否自启动。将求出的驱动方程，代入 JK 触发器的特性方程得到该电路的状态方程为

$$Q_2^{n+1} = Q_1^n Q_0^n \overline{Q}_2^n$$

$$Q_1^{n+1} = \overline{Q}_0^n \overline{Q}_1^n + \overline{Q}_0^n Q_1^n = Q_0^n \oplus Q_1^n$$

$$Q_0^{n+1} = \overline{Q}_2^n \overline{Q}_0^n$$

设电路的初始状态为 $Q_2Q_1Q_0=101$，代入上面的状态方程，可求出新状态为“010”；当初始状态为 $Q_2Q_1Q_0=110$，代入上面的状态方程，可求出新状态为“010”；初始状态为 $Q_2Q_1Q_0=111$，代入上面的状态方程，可求出新状态为“000”，故所设计的电路能够自启动。

(5) 画电路。根据驱动方程和输出方程可以画出电路，参考电路如图 6.2.1 所示。

由例 6.2.1 可知，本设计实现了一个同步五进制加法计数器。

【例 6.5.2】 试设计一个可控的模值为 3 的同步计数器。其控制条件为：当控制信号×为 0 时停止计数，当×为 1 时模 3 加法计数。

解　显然这是一个 Mealy 型的同步时序电路的设计。具体设计步骤如下。

(1) 分析逻辑功能要求，设置状态，建立原始状态图。显然，模值为 3 的计数器是指“逢三进一”的计数器，它的计数范围为 0～2，共 3 个状态。计数器有一个输入端和一个输出端，输入端用于输入计数脉冲，输出端输出进位信号；此外，还有一个计数控制端。

令 S_0 为电路的初始状态，当控制信号有效时，输入 1 个计数脉冲后进入到状态 S_1，输入 2 个计数脉冲后进入到状态 S_2，当输入 3 个计数脉冲后计数器返回到初始状态 S_0，同时向高位输出一个进位信号(即 $Y=1$)。根据逻辑功能要求，可画出如图 6.5.5 所示的原始状态图。

(2) 状态分配。原始状态图中不存等价状态，故可直接进行状态分配。原始状态图中共有 3 个状态，所以，选取两个触发器，其状态分配 $S=Q_2Q_1$，令 $S_0=00$，$S_1=01$，$S_2=10$，则可画出如图 6.5.6 所示的状态分配后的状态图。

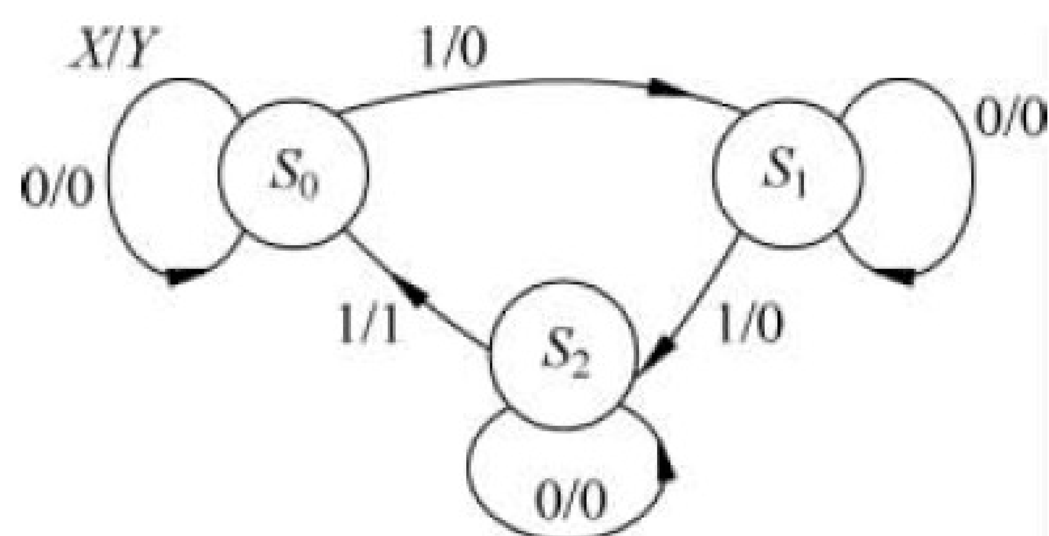


图 6.5.5　例 6.5.2 的图 1

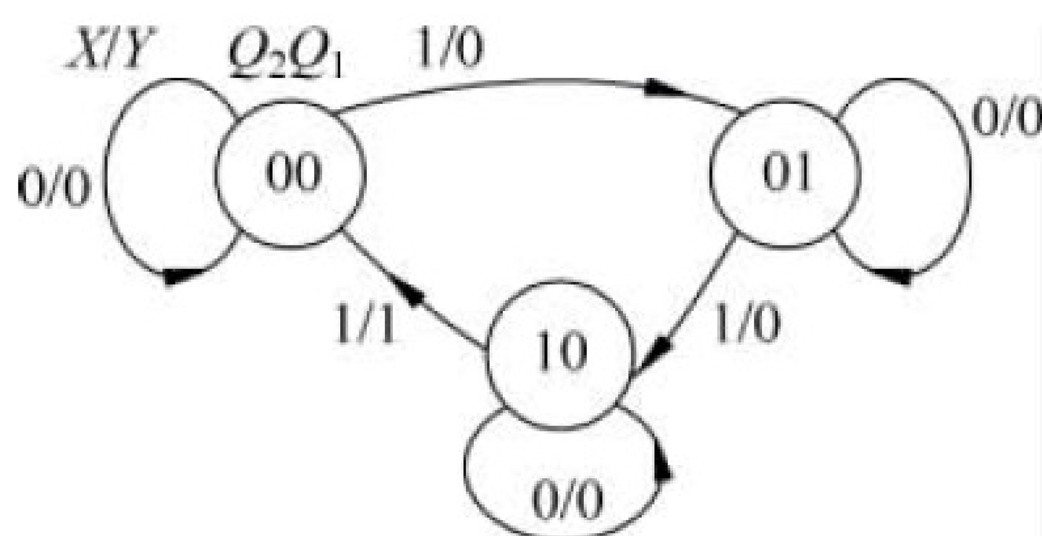


图 6.5.6　例 6.5.2 的图 2

(3) 确定触发器类型，求出触发器的驱动方程和电路的输出方程。

① 首先写出状态转换真值表(选 JK 触发器)。以 X、Q_2^n、Q_1^n作为自变量，Q_2^{n+1}、Q_1^{n+1}、

J_2、K_2、J_1、K_1、Y 为函数可列出如表 6.5.2 所示的表格。表中的第 1 列和第 2 列以及第 5 列由图 6.5.6 的状态图转换得到；表中的 J_2K_2、J_1K_1列由 JK 触发器的激励表写出。于是就得到如表 6.5.2 所示的状态转换真值表。

表 6.5.2 状态转换真值表

X Q_2^n Q_1^n	$Q_2^{n+1}Q_1^{n+1}$	J_2 K_2	J_1 K_1	Y
0 0 0	0 0	0 ×	0 ×	0
0 0 1	0 1	0 ×	× 0	0
0 1 0	1 0	× 0	0 ×	0
0 1 1	× ×	× ×	× ×	×
1 0 0	0 1	0 ×	1 ×	0
1 0 1	1 0	1 ×	× 1	0
1 1 0	0 0	× 1	0 ×	1
1 1 1	× ×	× ×	× ×	×

② 求出驱动方程、输出方程。由表 6.5.2 分别经过如图 6.5.7 的卡诺图可求出驱动方程和输出方程。

由图 6.5.7(a)、图 6.5.7(b)可求出 $J_2=XQ_1$、$K_2=X$；

由图 6.5.7(c)、图 6.5.7(d)求出 $J_1=X\bar{Q}_2$、$K_1=X$；

由图 6.5.7(e)求出输出 $Y=XQ_2$。

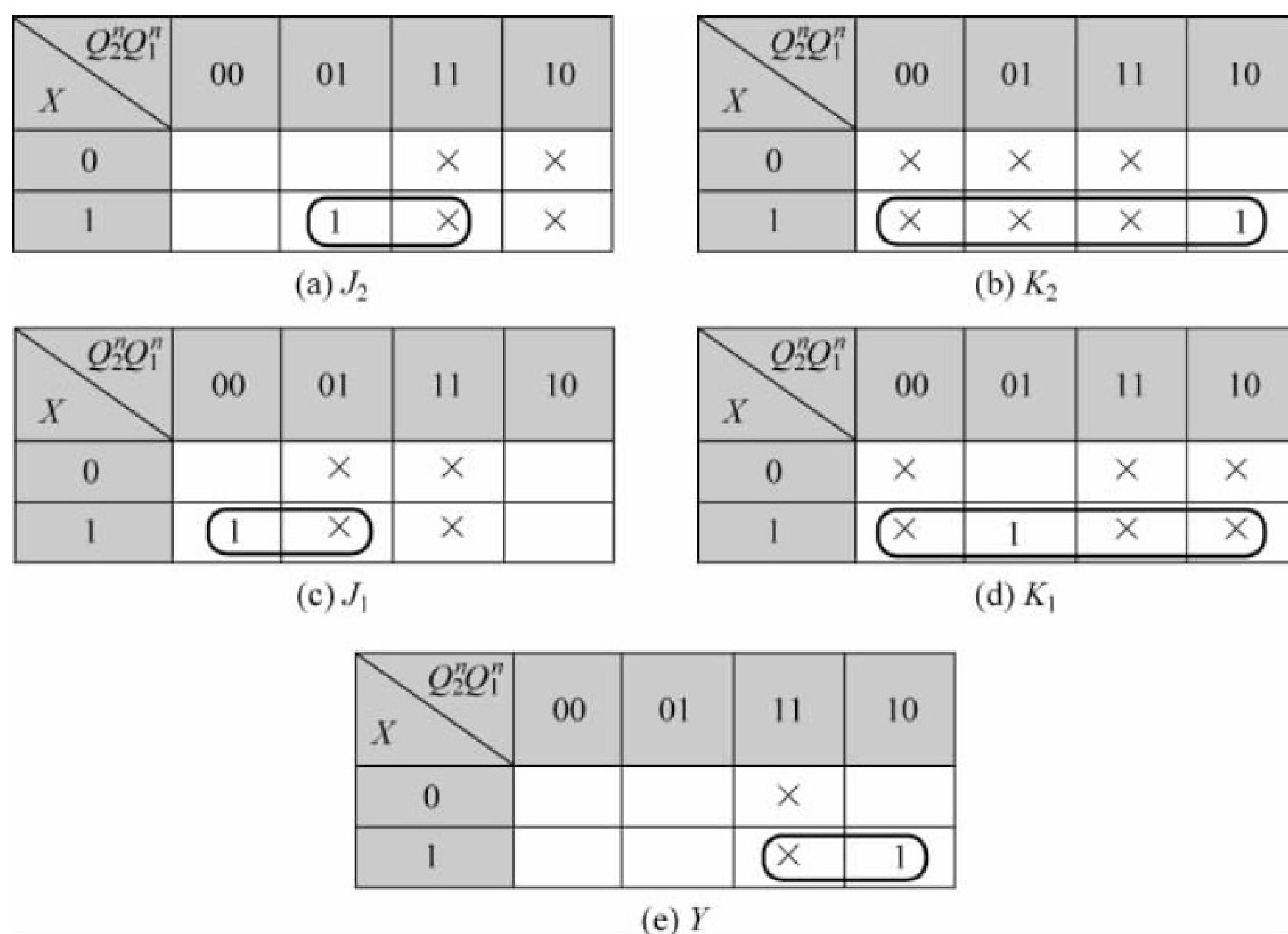


图 6.5.7 例 6.5.2 的图 3

(4) 检查自启动。由于所设计的电路存在无效状态，即 $Q_2^nQ_1^n=11$，所以需检查所设计的电路能否自启动(这一步骤也可以放在后面进行)。

将上述求出的驱动方程代入 JK 触发器的特性方程，可得到该电路的状态方程，即

$$Q_1^{n+1} = X\overline{Q}_2^n\overline{Q}_1^n + \overline{X}Q_1^n$$

$$Q_2^{n+1} = XQ_1^n\overline{Q}_2^n + \overline{X}Q_2^n$$

设电路的初始状态为 $Q_2^nQ_1^n=11$，当 CP 脉冲到来时，代入上面的状态方程，可求出当 $X=0$ 时，新状态为“11”；当 $X=1$ 时，新状态为“00”；故所设计的电路能够自启动。

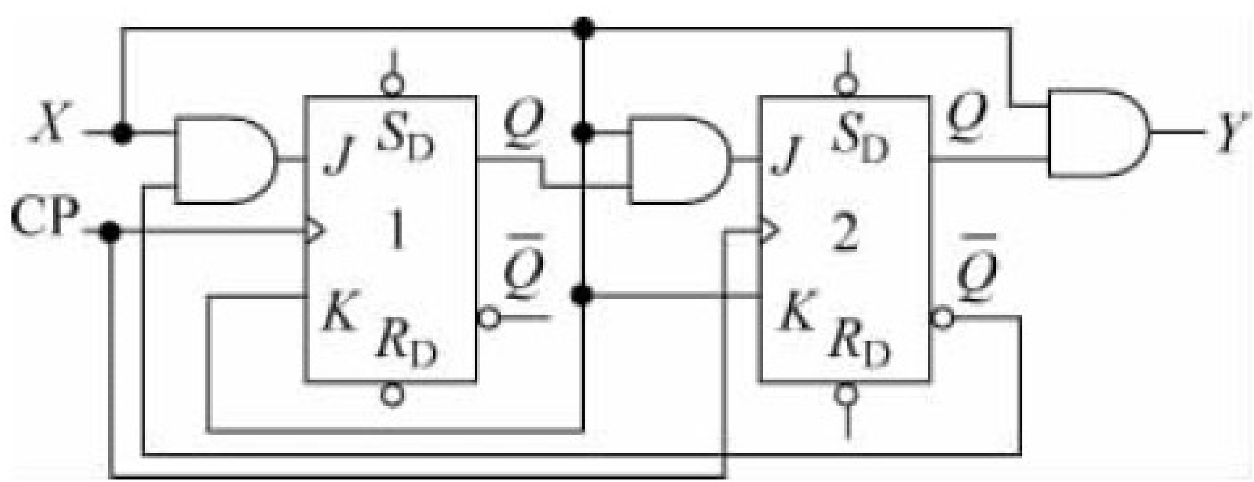


图 6.5.8　例 6.5.2 的图 4

（5）画电路。根据驱动方程，输出方程可以画出如图 6.5.8 所示的可控模值为 3 的计数器电路图。

6.5.2　用中规模时序电路芯片实现实际逻辑问题的方法

常用中规模器件设计实现时序逻辑电路。前面介绍的利用 74LS161 等集成电路芯片实现自然规律计数器的方法就是用中规模器件设计实现时序逻辑电路的主要方式之一。



当然，实际的时序逻辑电路状态变化往往不是自然计数规律，可利用这些设计好的、按自然计数规律变化的计数器去实现实际的时序逻辑电路状态变化图。事实上，一个八进制的自然规律计数器可实现任意的 8 个状态循环的时序电路，实现方法如下：

八进制的自然规律计数器可实现 8 个状态循环，计数器输出与要求输出不一致，可将计数器输出当作中间输入，利用组合电路，将其变换为要求的 8 个状态即可，可结合下面的实例来理解。

【例 6.5.3】 请设计在时钟作用下按照如表 6.5.3 所示顺序发生状态转换的灯光控制逻辑。表中的 1 表示灯“亮”，0 表示灯“灭”。

解　（1）由表 6.5.3 知，该灯光控制逻辑 8 个时钟完成一次状态循环，可利用 74161 实现该功能。

（2）74161 内部状态按照自然规律循环，与表 6.5.3 不吻合，可将 74161 的 $Q_2Q_1Q_0$ 作为输入，用 74138 结合门电路实现如表 6.5.3 所示的红（R）、黄（Y）、绿（G）输出。该逻辑函数的真值表如表 6.5.4 所示。

表 6.5.3　例 6.5.3 的原始状态表

CP	红	黄	绿
0	0	0	0
1	1	0	0
2	0	1	0
3	0	0	1
4	1	1	1
5	0	0	1
6	0	1	0
7	1	0	0
8	0	0	0

表 6.5.4　例 6.5.3 的真值表

Q_2	Q_1	Q_0	红	黄	绿
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	1	0
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	0	0	1
1	1	0	0	1	0
1	1	1	1	0	0

令 74LS138 译码器的地址端分别为 $A_2=Q_2$，$A_1=Q_1$，$A_0=Q_0$，则它的输出就是表 6.5.4 中的 $\overline{m}_0 \sim \overline{m}_7$，有

$$R=m_1+m_4+m_7=\overline{\overline{m}_1 \cdot \overline{m}_4 \cdot \overline{m}_7}=\overline{\overline{Y}_1 \cdot \overline{Y}_4 \cdot \overline{Y}_7}$$

$$Y=m_2+m_4+m_6=\overline{\overline{m}_2 \cdot \overline{m}_4 \cdot \overline{m}_6}=\overline{\overline{Y}_2 \cdot \overline{Y}_4 \cdot \overline{Y}_6}$$

$$G=m_3+m_4+m_5=\overline{\overline{m}_3 \cdot \overline{m}_4 \cdot \overline{m}_5}=\overline{\overline{Y}_3 \cdot \overline{Y}_4 \cdot \overline{Y}_5}$$

可画出电路如图 6.5.9 所示。

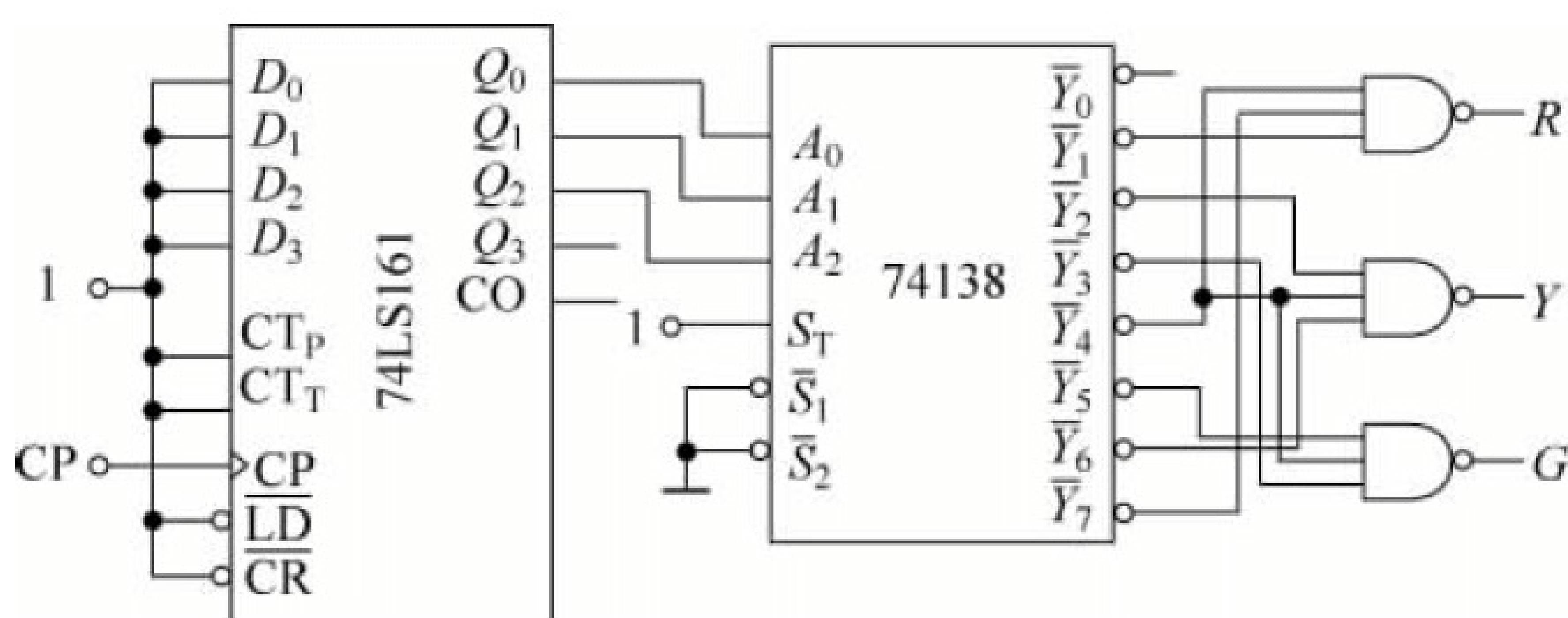


图 6.5.9 例 6.5.3 的图

【例 6.5.4】 请设计一个 4 路输出的顺序正脉冲发生器。

解 前面介绍的如图 6.4.66 所示环形计数器构成了一个 4 路输出的顺序正脉冲发生器，也可利用计数器结合译码器实现顺序脉冲发生器。

(1) 四进制计数器的实现。4 路输出的顺序正脉冲发生器每 4 个脉冲循环一次，可理解为四进制计数器。74LS161 为十六进制计数器，其低 2 位 Q_1Q_0 为四进制计数器。

(2) 4 路输出的顺序正脉冲状态变化规律的实现。74LS161 低 2 位 Q_1Q_0 实现的四进制计数器按照自然规律循环，与顺序正脉冲状态变化规律不吻合，可将 74LS161 的 Q_1Q_0 作为输入，顺序正脉冲的 4 路输出 $O_0O_1O_2O_3$ 作为输出，可做出真值表如表 6.5.5 所示。

表 6.5.5 例 6.5.4 的真值表

Q_1	Q_0	O_0	O_1	O_2	O_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

(3) 用 74LS138 结合门电路实现 4 路输出的顺序正脉冲状态变化规律。

令 74LS138 译码器的地址端分别为 $A_2=0$，$A_1=Q_1$，$A_0=Q_0$，则它的输出 $\overline{Y}_0 \sim \overline{Y}_3$ 就是表 6.5.5 中的 $\overline{m}_0 \sim \overline{m}_3$，有

$$O_0=m_0=\overline{\overline{m}}_0=\overline{\overline{Y}}_0$$

$$O_1=m_1=\overline{\overline{m}}_1=\overline{\overline{Y}}_1$$

$$O_2=m_2=\overline{\overline{m}}_2=\overline{\overline{Y}}_2$$

$$O_3=m_3=\overline{\overline{m}}_3=\overline{\overline{Y}}_3$$

可画出电路如图 6.5.10 所示。

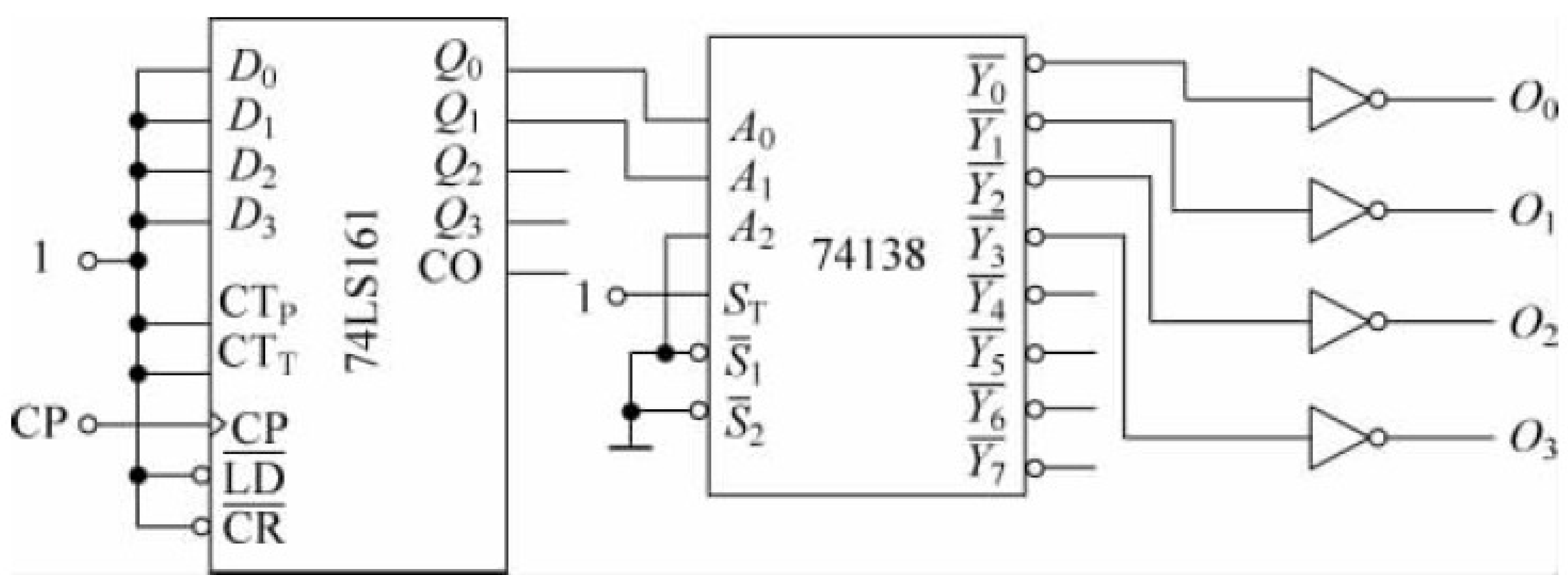


图 6.5.10 例 6.5.4 的图

【例 6.5.5】 请设计一个“1110”序列发生器。

解 (1) 四进制计数器的实现。“1110”序列发生器只有 1 个输出，但在每 4 个输入脉冲激励下输出将循环一次，可理解为四进制计数器。74LS161 为十六进制计数器，其低 2 位 Q_1Q_0 为四进制计数器。

(2) “1110”序列输出的实现。可利用数据选择器实现“1110”序列的输出。如令四选一数据选择器的 $D_0=1$，$D_1=1$，$D_2=1$，$D_3=0$。利用四进制计数器控制四选一数据选择器的地址端，使数据选择器的输出 Y 轮流输出 $D_0\sim D_3$ 的值，则数据选择器的输出为“1110”序列。

74LS151 为八选一数据选择器，令四进制计数器的 Q_1 接 74LS151 的 A_1、Q_0 接 A_0，将 A_2 接 0，则 74LS151 只有低 4 位数据选择进入选择器。令 74LS151 的 $D_0=1$，$D_1=1$，$D_2=1$，$D_3=0$ 可实现“1110”序列的输出。

可画出电路如图 6.5.11 所示。

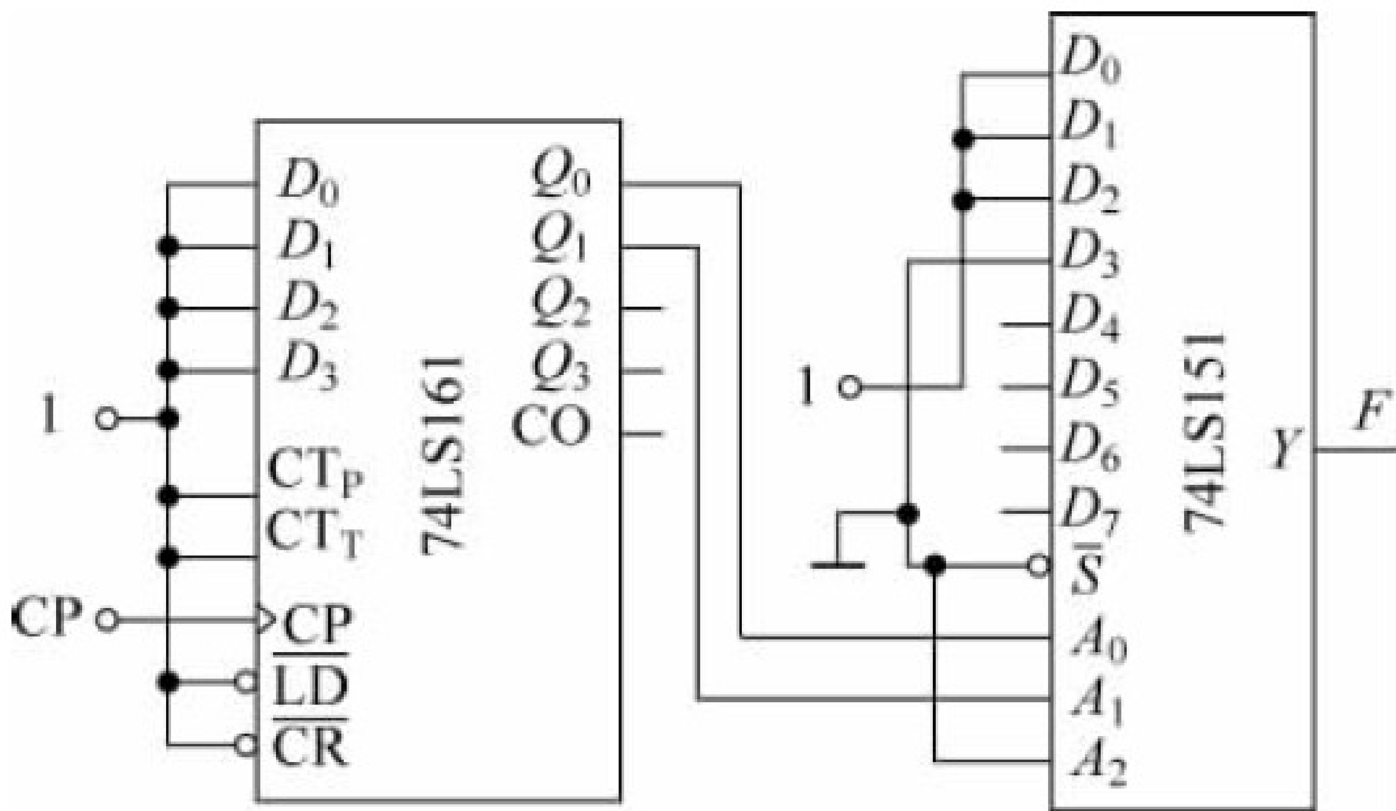


图 6.5.11 例 6.5.5 的图

复习与思考

6.5.1 用 D 触发器实现例 6.5.1。

6.5.2 例 6.5.5 中，令 74LS151 的 $D_4=1$，$D_5=1$，$D_6=1$，$D_7=0$，其余各数据输入端接地，如何实现“1110”序列发生器？

习题

6-1 填空题

1. 时序逻辑电路可以用________、存储电路的________、存储电路的________等与输入、电路的现态关系的三组关系方程来描述。

2. 分析异步时序电路比分析同步时序电路复杂。分析各触发器状态变化规律时，只有那些有________的触发器才需要用状态方程去________，而没有________的触发器则保持________不变。

3. 移位寄存器的右移功能可用________来描述。移位寄存器广泛用于数学运算。移位寄存器中________ 1 位等于数学上左移 1 位，数值上等于________。

4. 74LS195 为________、________的右移移位寄存器。Q_0的值依照外部输入 J、$\overline{K}$ 的值按照________发生变化。

5. 一个计数器所能够记入计数脉冲的数目，称为计数器的________、计数长度或计数器的________。八进制计数器，每________个时钟脉冲，输出 Y 输出________个脉冲，因此，也称________分频器。

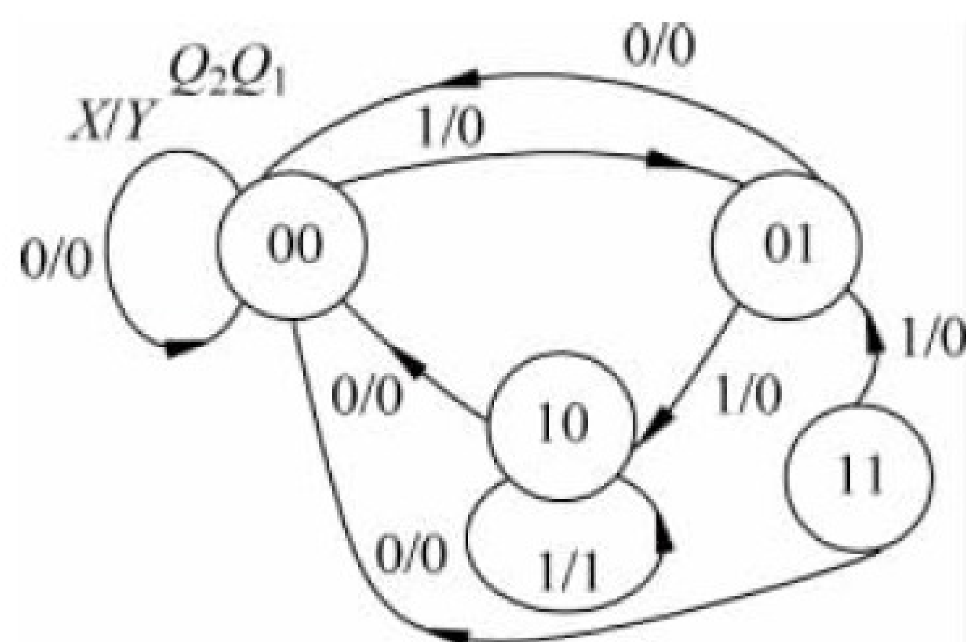
6. 74LS160 是一个具有________、________、可以保持状态不变的十进制同步加法计数器。74LS162 与 74LS160 的区别是 74LS162 采用________清零方式，即当$\overline{\mathrm{CR}}=0$ 时，还需要 CP 脉冲________到来时，计数器才被清零。

7. 集成计数器清零、置数有两种工作方式：________、________。所谓________工作方式，是指通过时钟触发器异步输入端($\overline{R}_D$ 端或 $\overline{S}_D$)实现清零或置数，而与________无关。

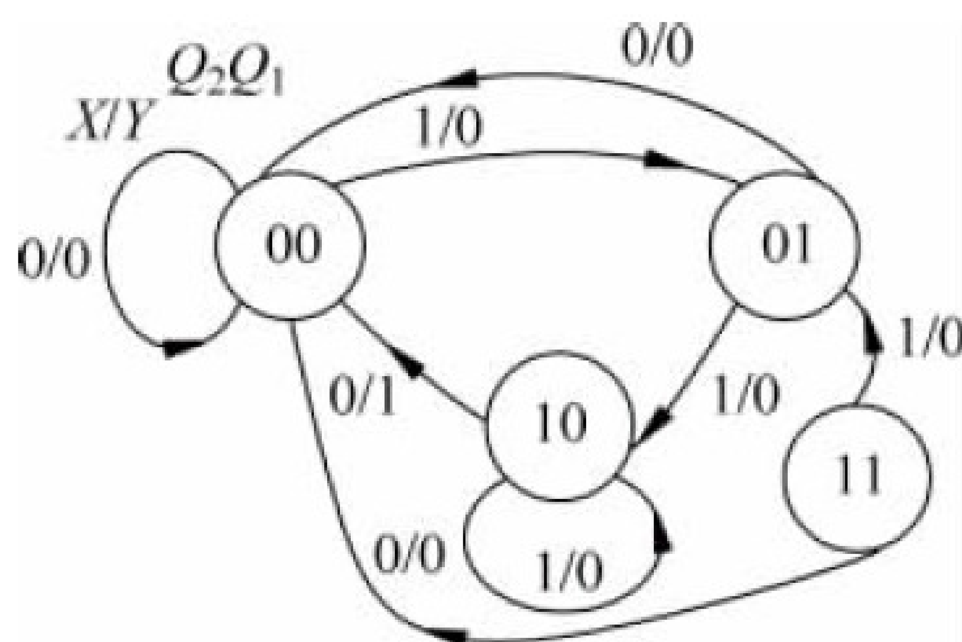
6-2 分析设计题

1. 试分析如题图 6.1 所示状态图所对应的状态转移真值表(图中，X 表示电路的输入，Y 表示电路的输出)及描述的逻辑功能。

2. 试分析如题图 6.2 所示状态图所对应的状态转移真值表(图中，X 表示电路的输入，Y 表示电路的输出)及描述的逻辑功能。

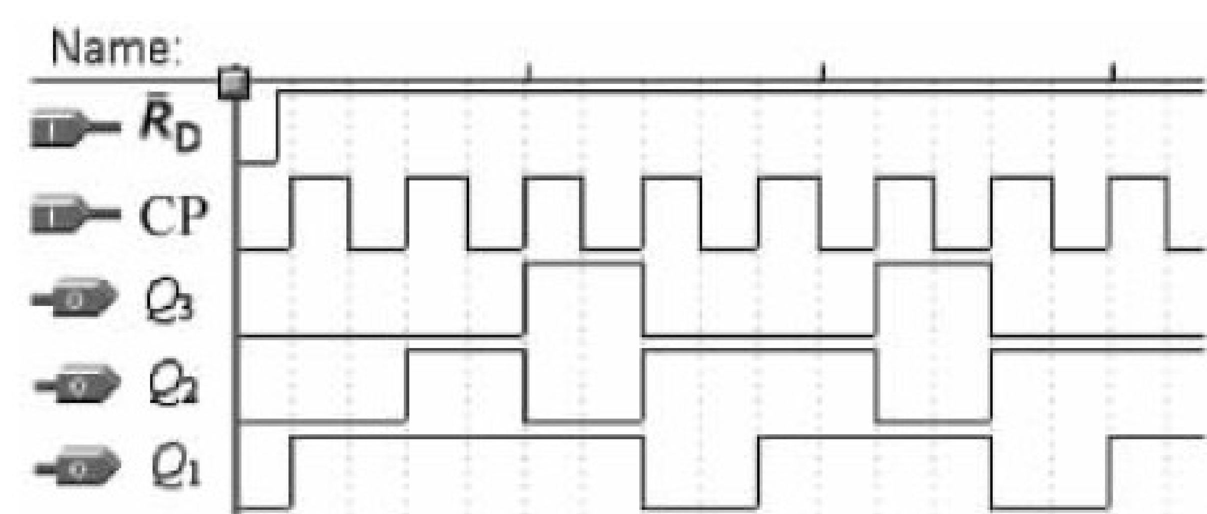


题图 6.1



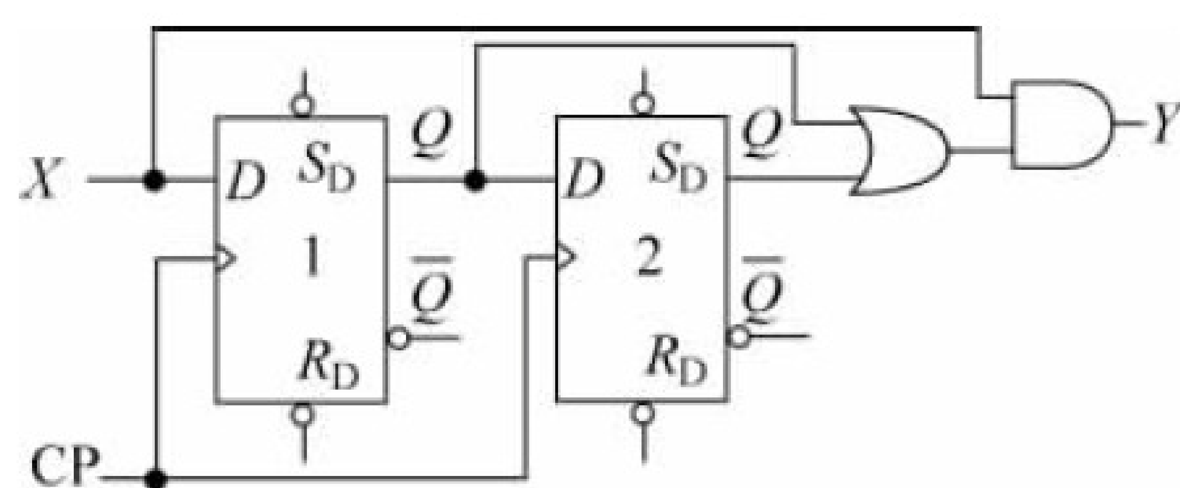
题图 6.2

3. 试分析如题图 6.3 所示时序图所对应的有效循环状态图及描述的逻辑功能。



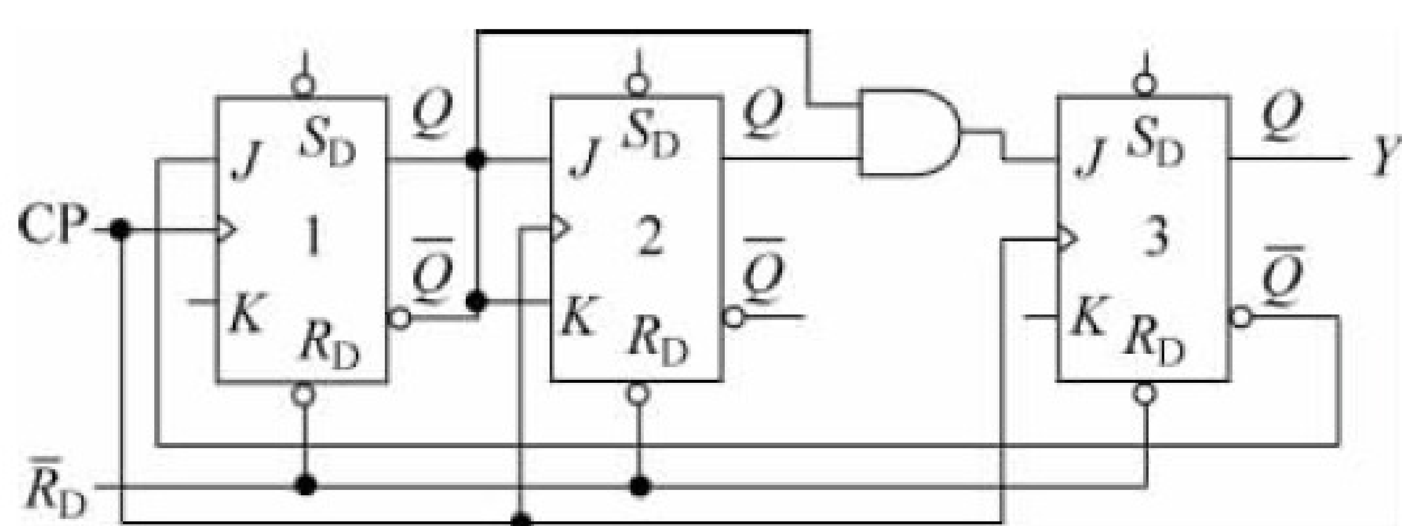
题图 6.3

4. 试写出如题图 6.4 所示电路的驱动方程、状态方程及输出方程，画出电路的状态图，说明电路的逻辑功能及其自启动状况。



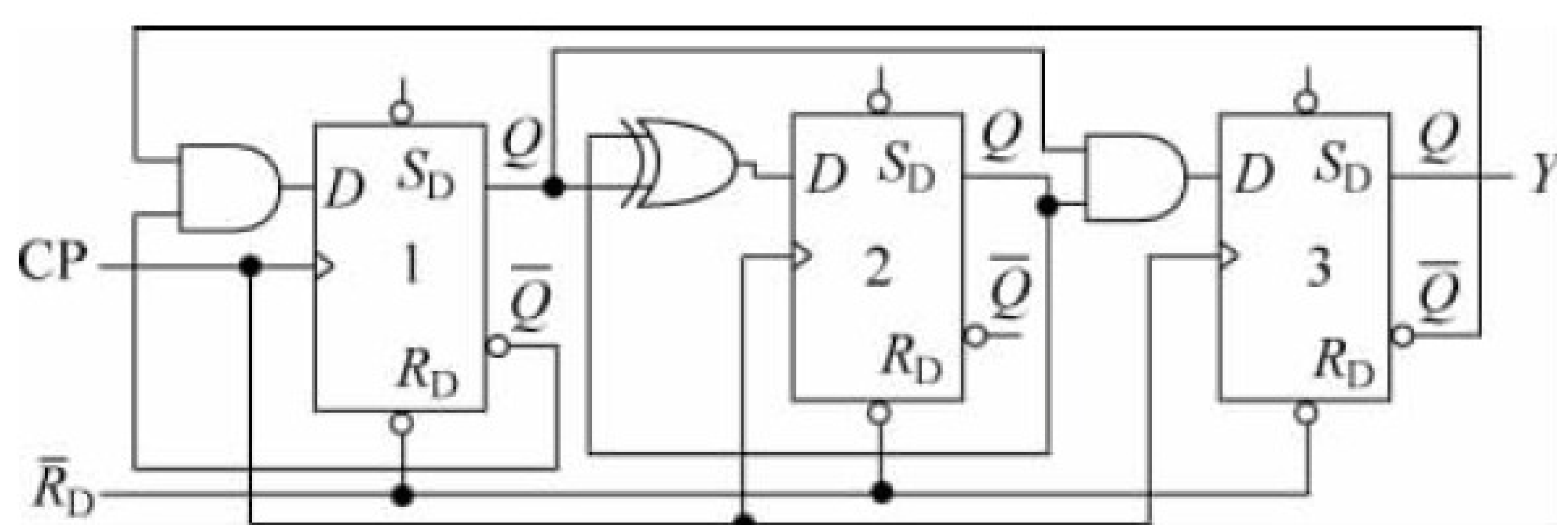
题图 6.4

5. 试写出如题图 6.5 所示电路的驱动方程、状态方程及输出方程，画出电路的状态图，说明电路能否自启动。



题图 6.5

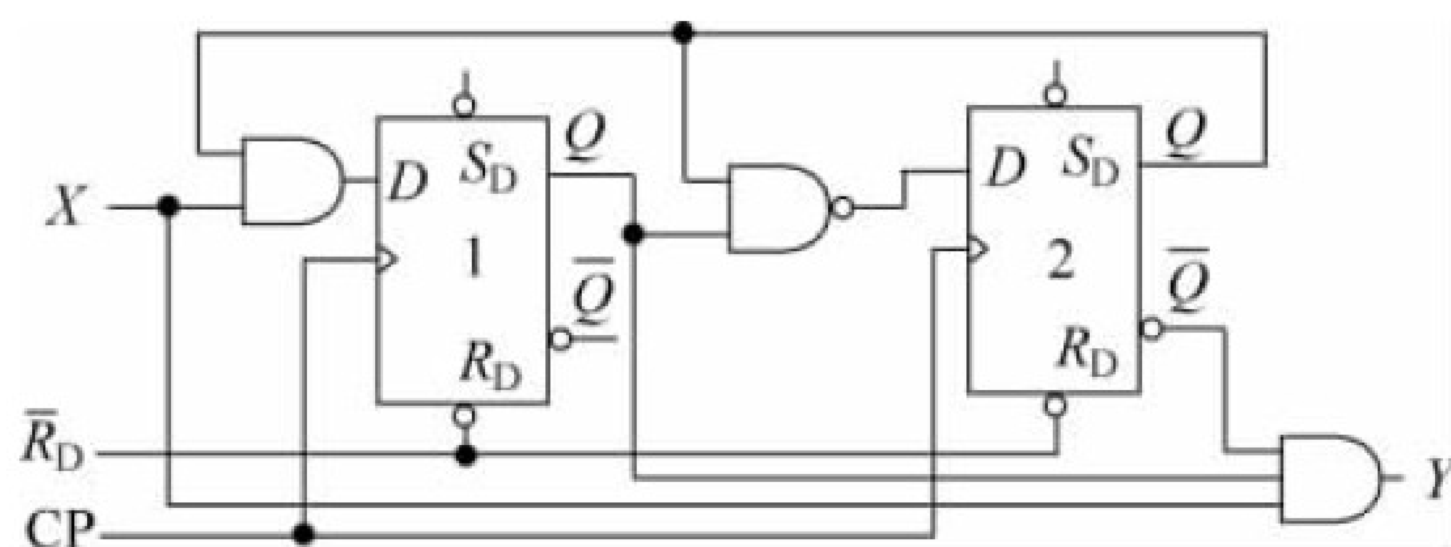
6. 试写出如题图 6.6 所示电路的驱动方程、状态方程及输出方程，画出电路的状态图，说明电路的逻辑功能及其自启动状况。



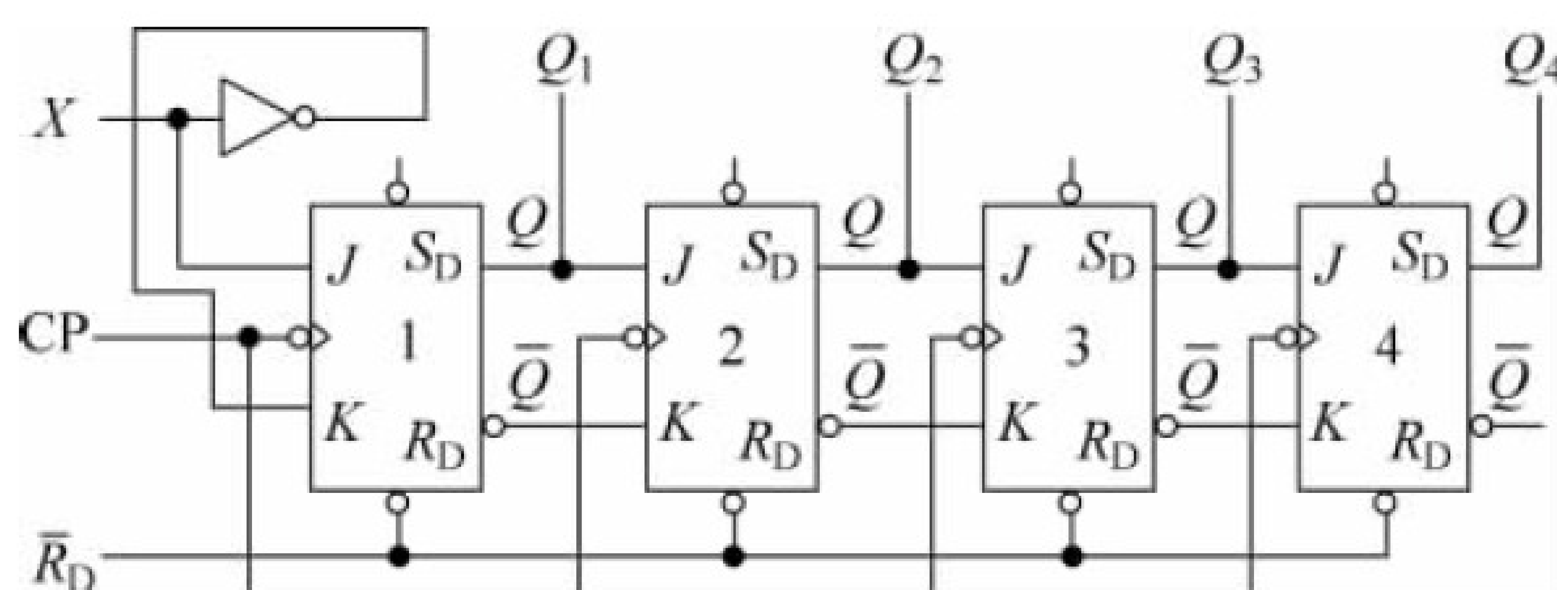
题图 6.6

7. 试写出如题图 6.7 所示电路的驱动方程、状态方程及输出方程，画出电路的状态图，说明电路的逻辑功能及其自启动状况。

8. 试写出如题图 6.8 所示电路的驱动方程、状态方程及输出方程，画出电路的状态图，说明电路的逻辑功能及其自启动状况。

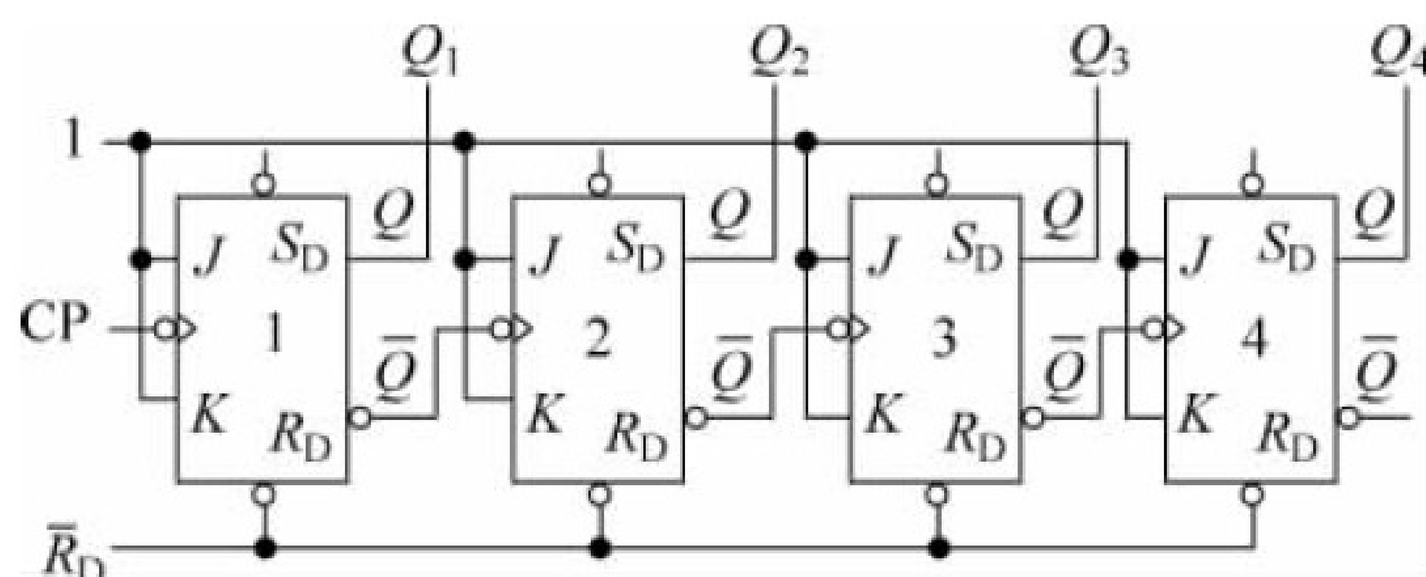


题图 6.7



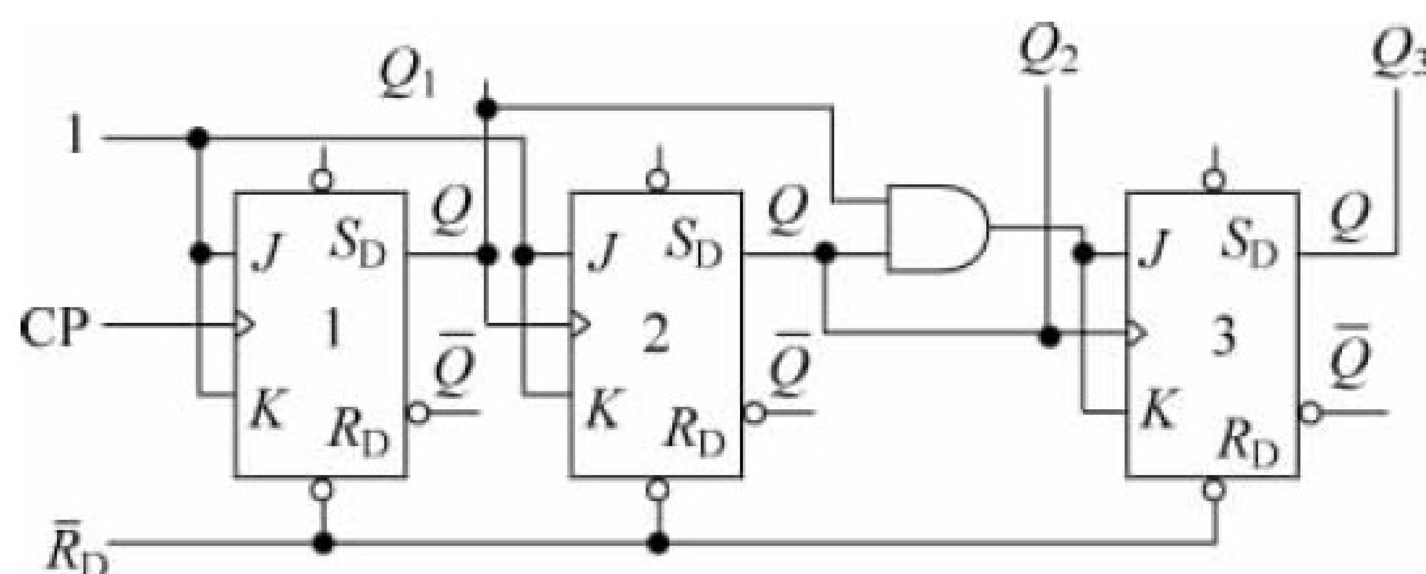
题图 6.8

9. 试写出如题图 6.9 所示电路(下降沿触发器)的驱动方程、状态方程及输出方程,画出电路的状态图,说明电路的逻辑功能及其自启动状况。



题图 6.9

10. 试写出如题图 6.10 所示电路(上升沿触发器)的驱动方程、状态方程及输出方程,画出电路的状态图,说明电路的逻辑功能及其自启动状况。

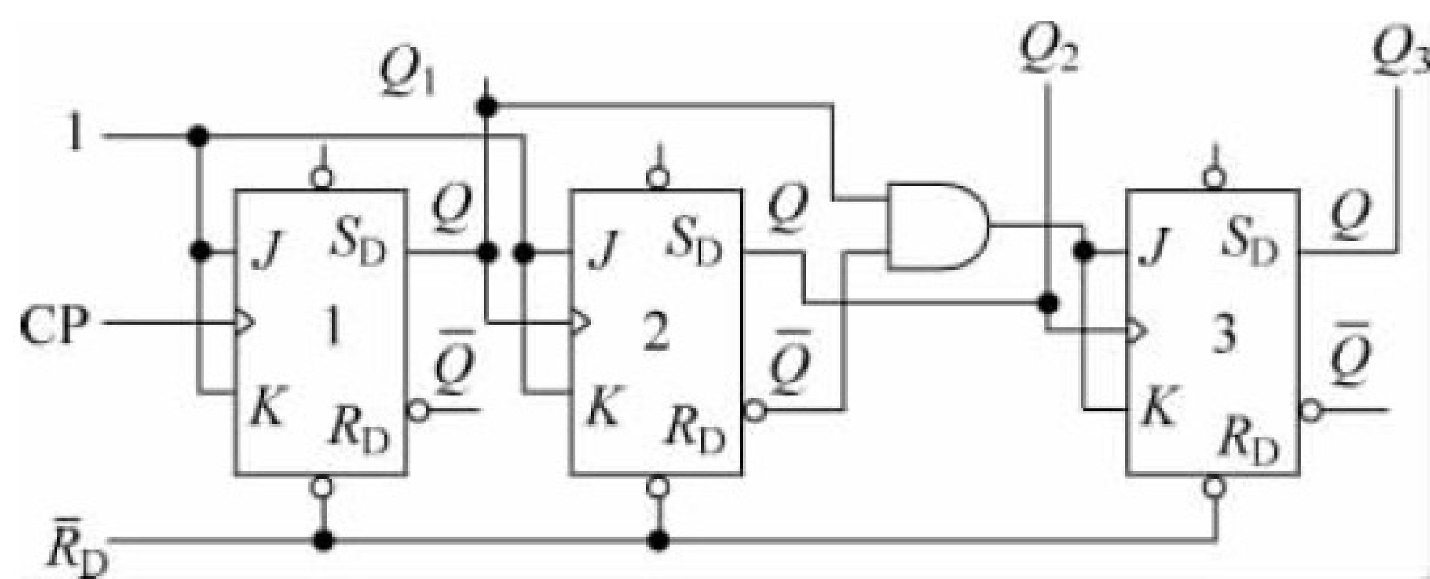


题图 6.10

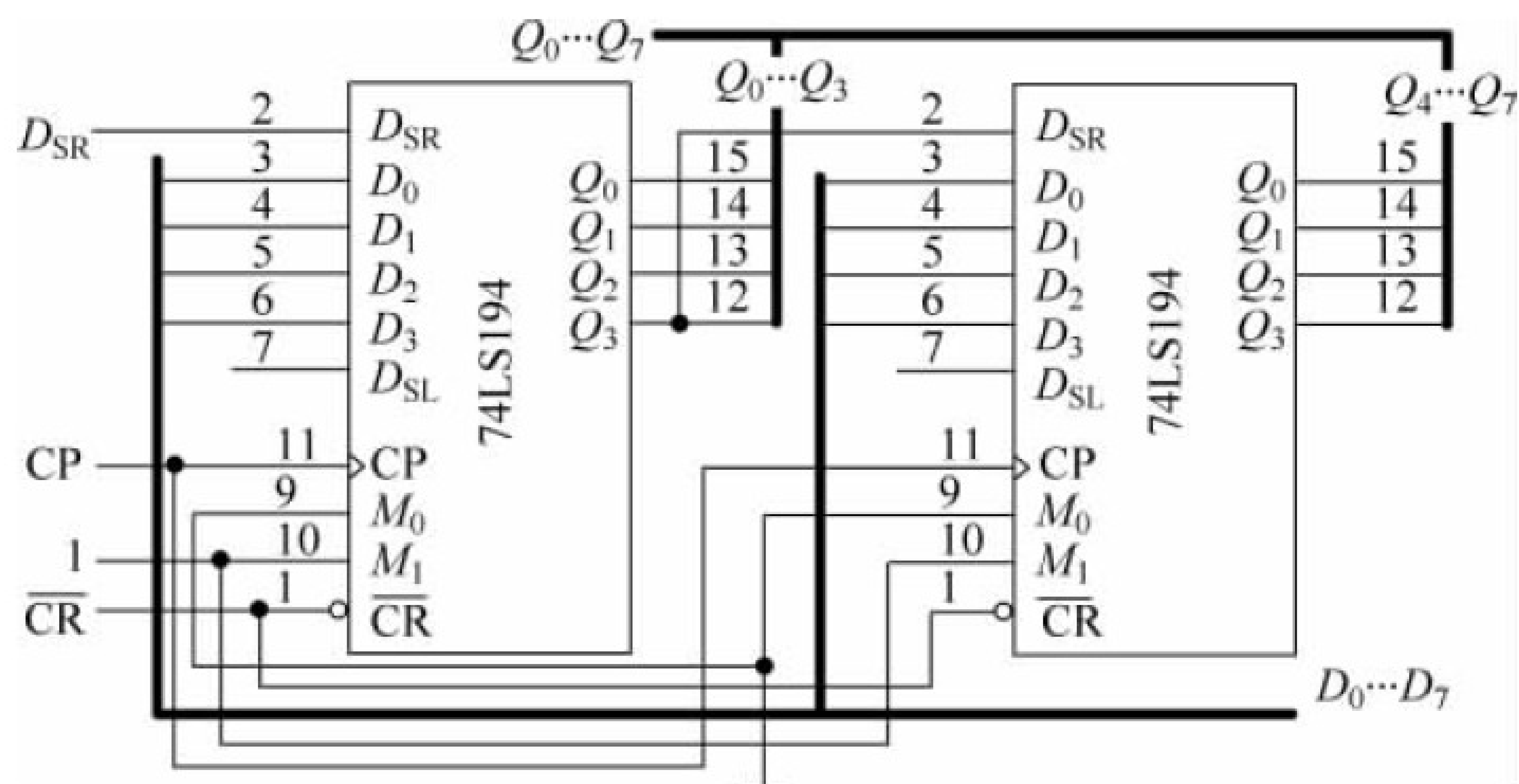
11. 试写出如题图 6.11 所示电路(上升沿触发器)的驱动方程、状态方程及输出方程,画出电路的状态图,说明电路的逻辑功能及其自启动状况。

12. 分析如题图 6.12 所示电路的逻辑功能。

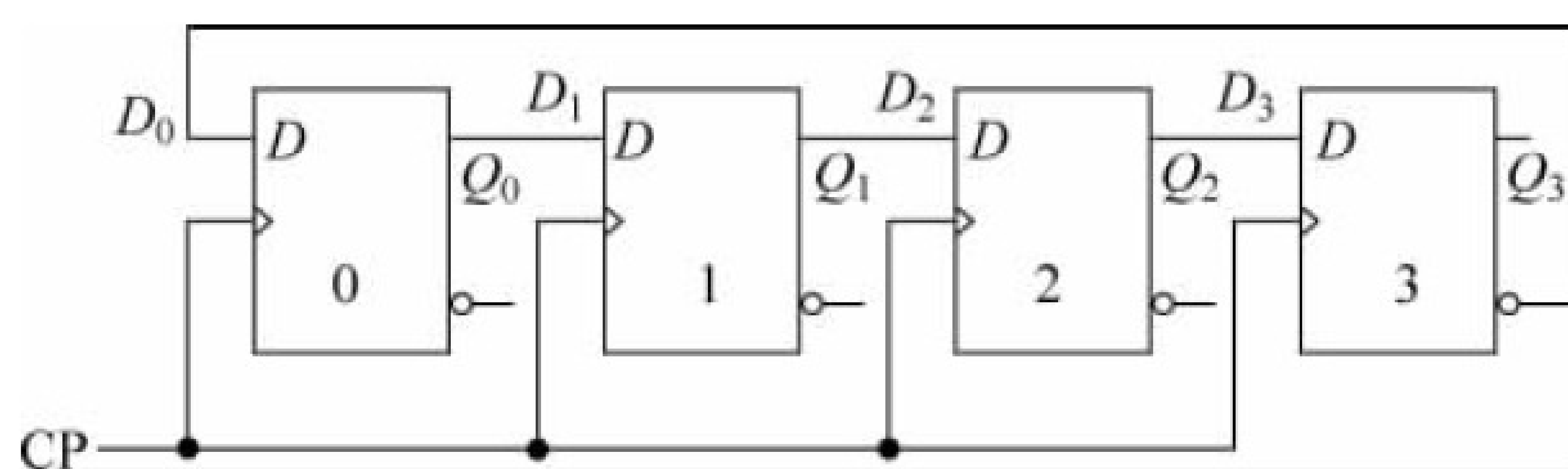
13. 如题图 6.13 所示电路各触发器初始状态为 0,分析其逻辑功能。



题图 6.11

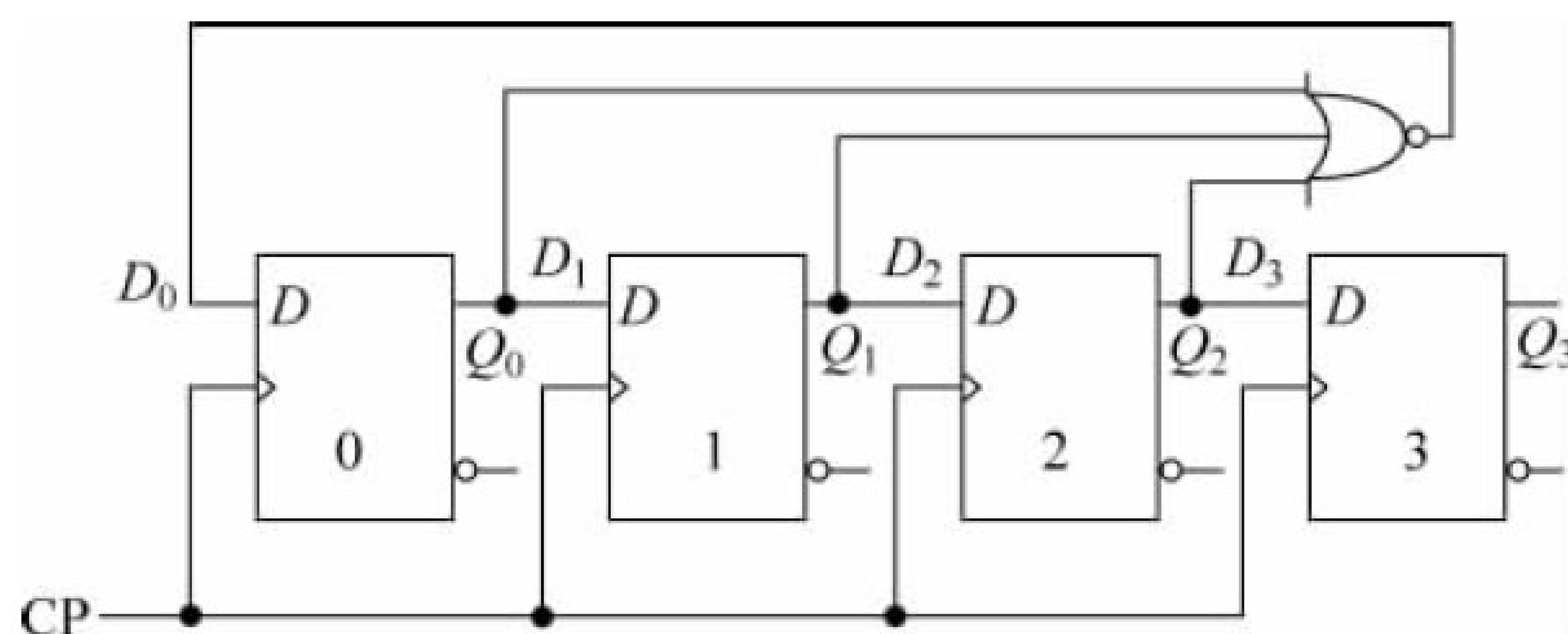


题图 6.12



题图 6.13

14. 分析如题图 6.14 所示电路的逻辑功能。



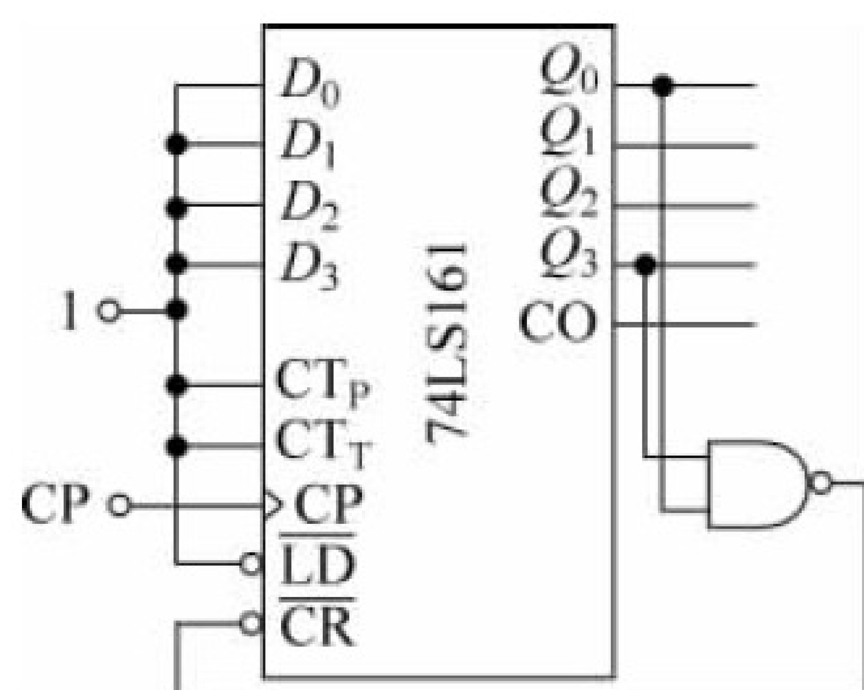
题图 6.14

15. 分析如题图 6.15 所示电路为多少进制计数器。

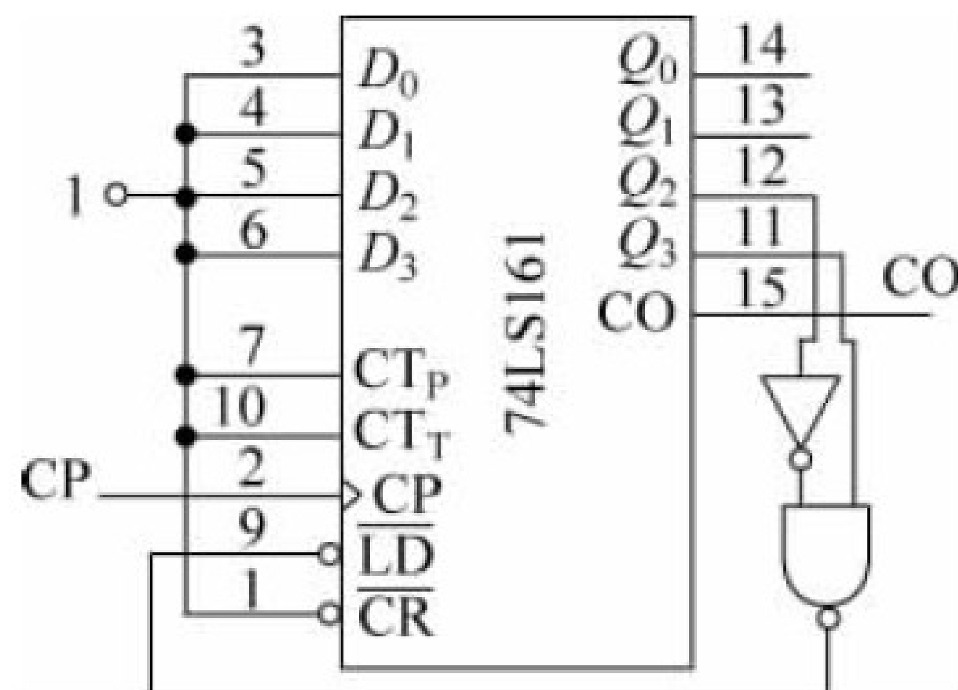
16. 试分析如题图 6.16 所示电路的逻辑功能。

17. 分析题图 6.17 所示电路为多少进制计数器。

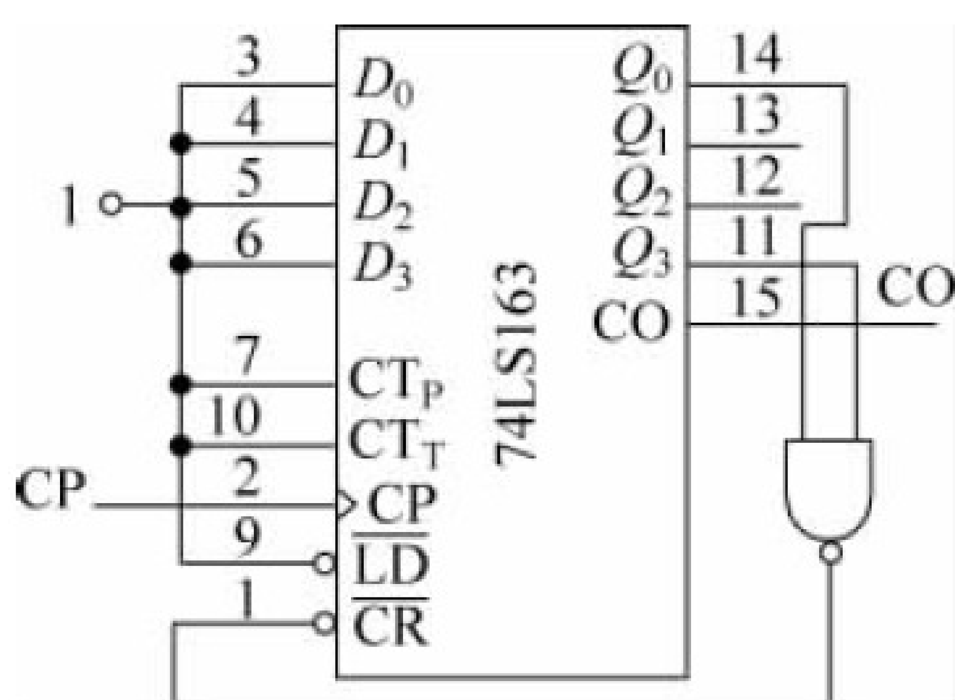
18. 试分析如题图 6.18 所示电路的逻辑功能。



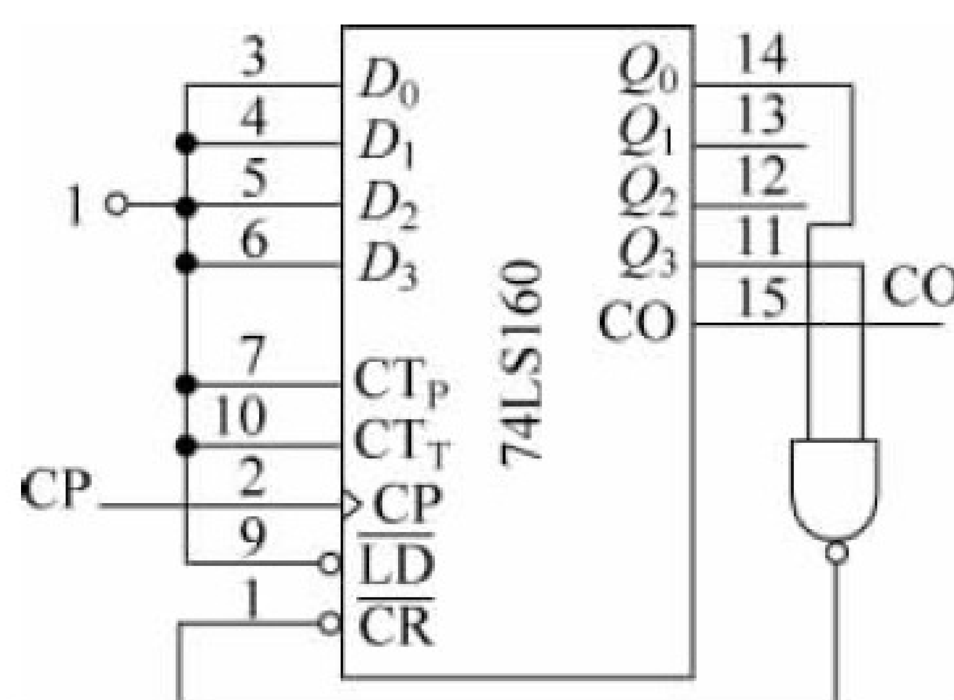
题图 6.15



题图 6.16

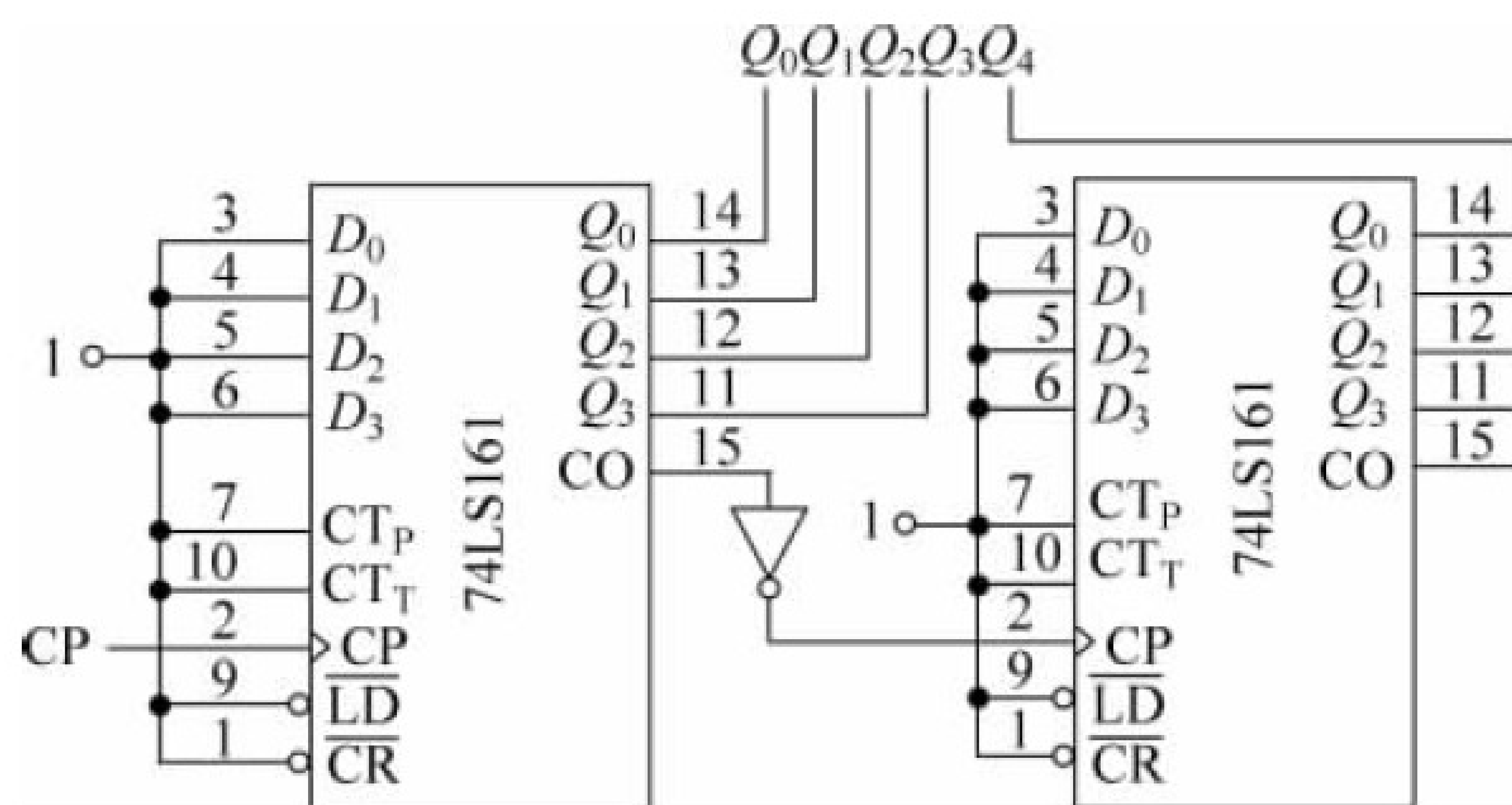


题图 6.17



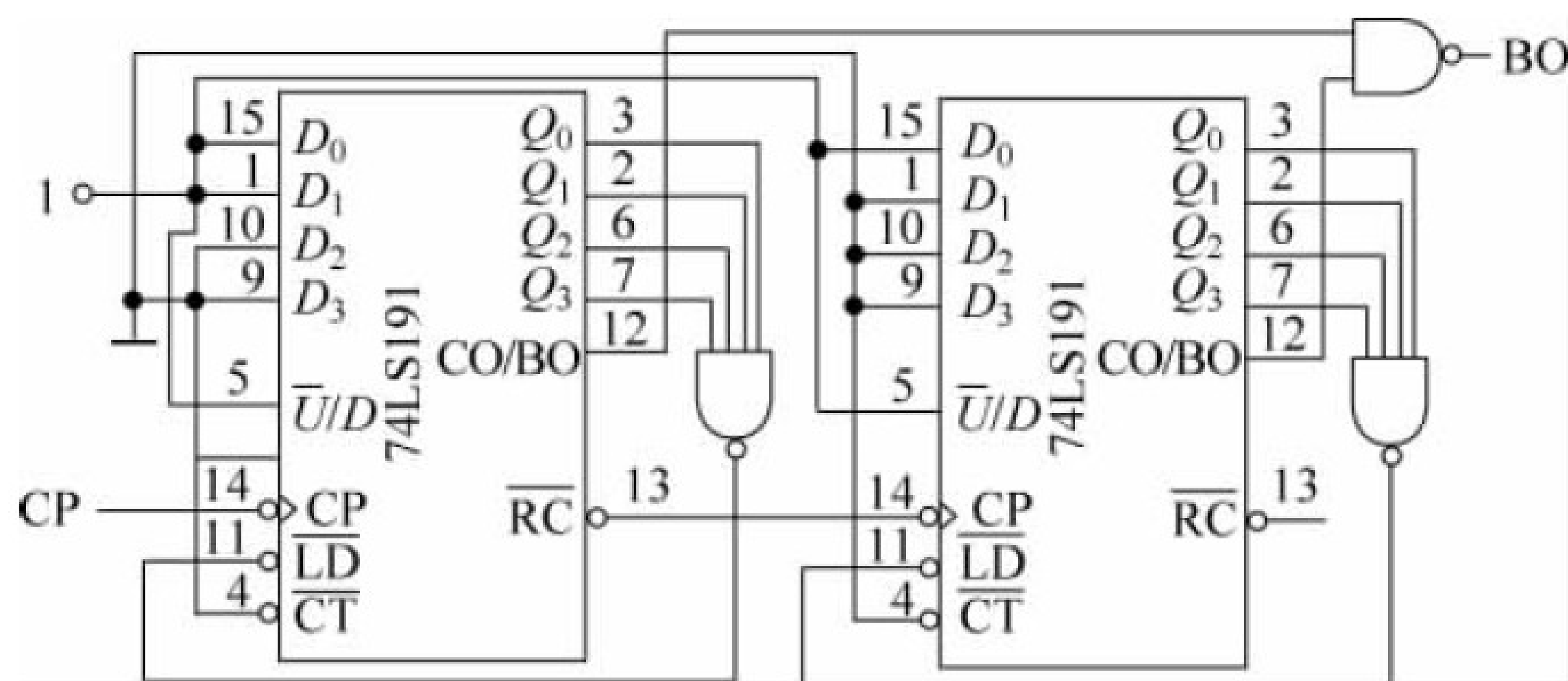
题图 6.18

19. 试分析如题图 6.19 所示电路中 $Q_4Q_3Q_2Q_1Q_0$ 构成几进制计数器。



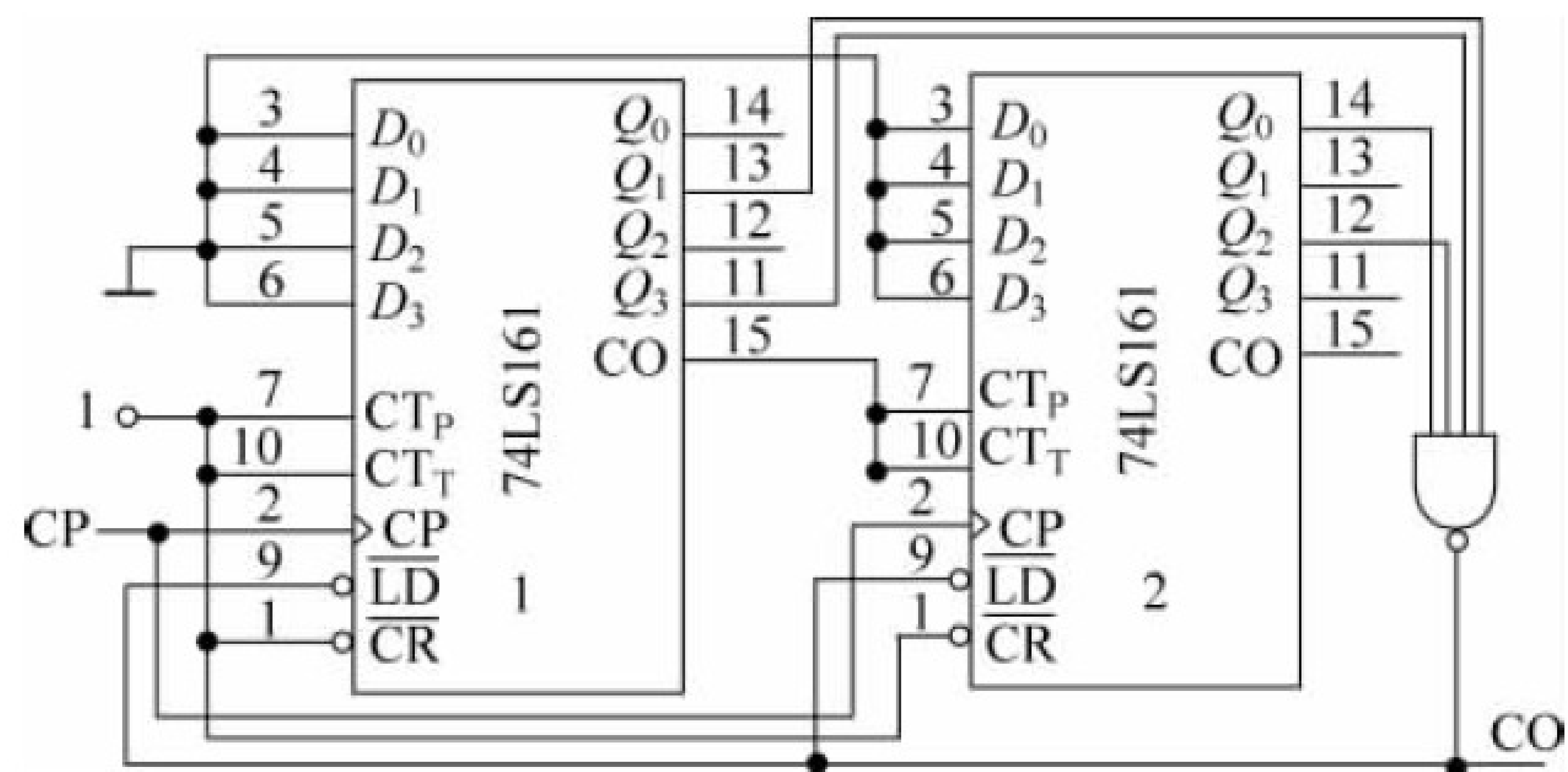
题图 6.19

20. 试分析如题图 6.20 所示电路的逻辑功能。



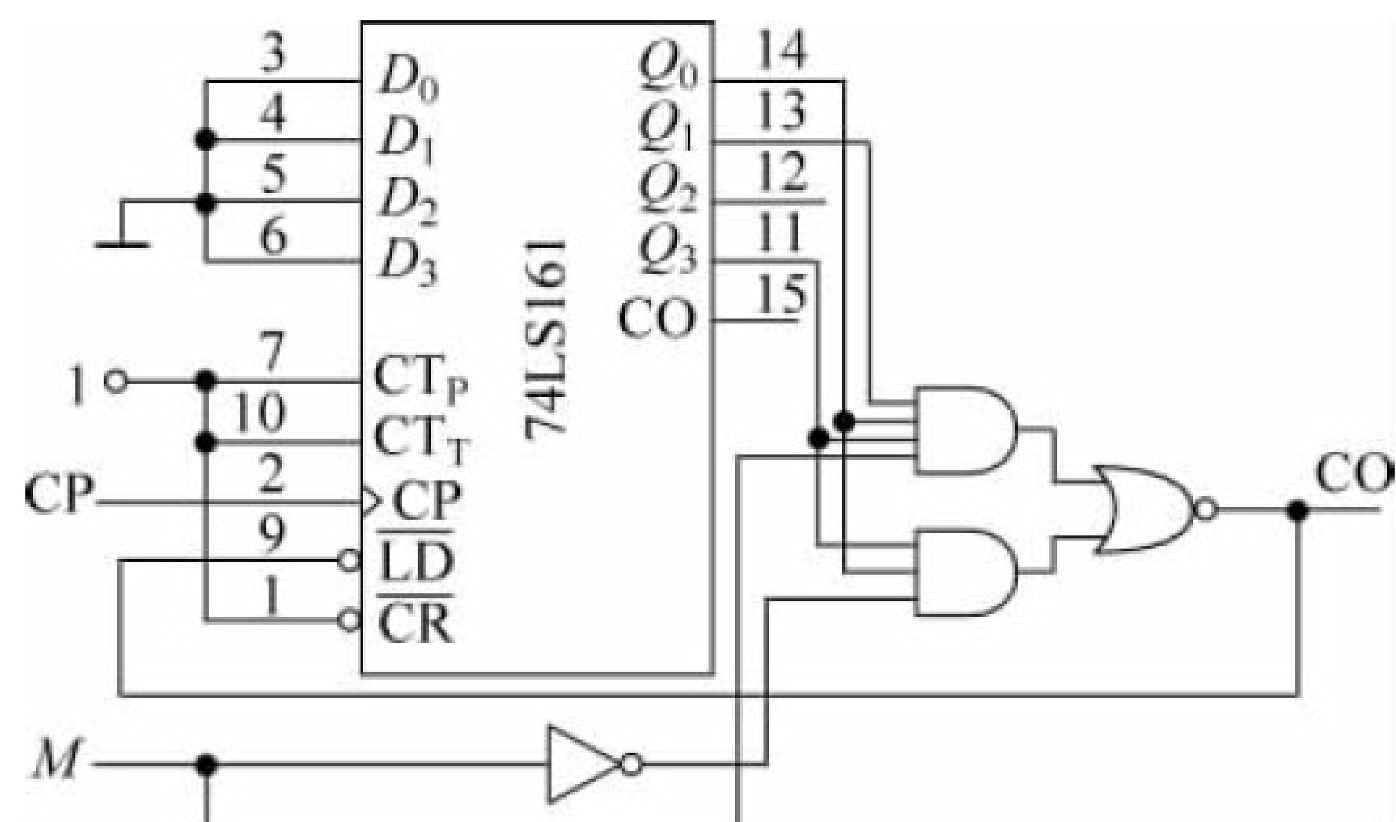
题图 6.20

21. 试分析如题图 6.21 所示电路的逻辑功能。



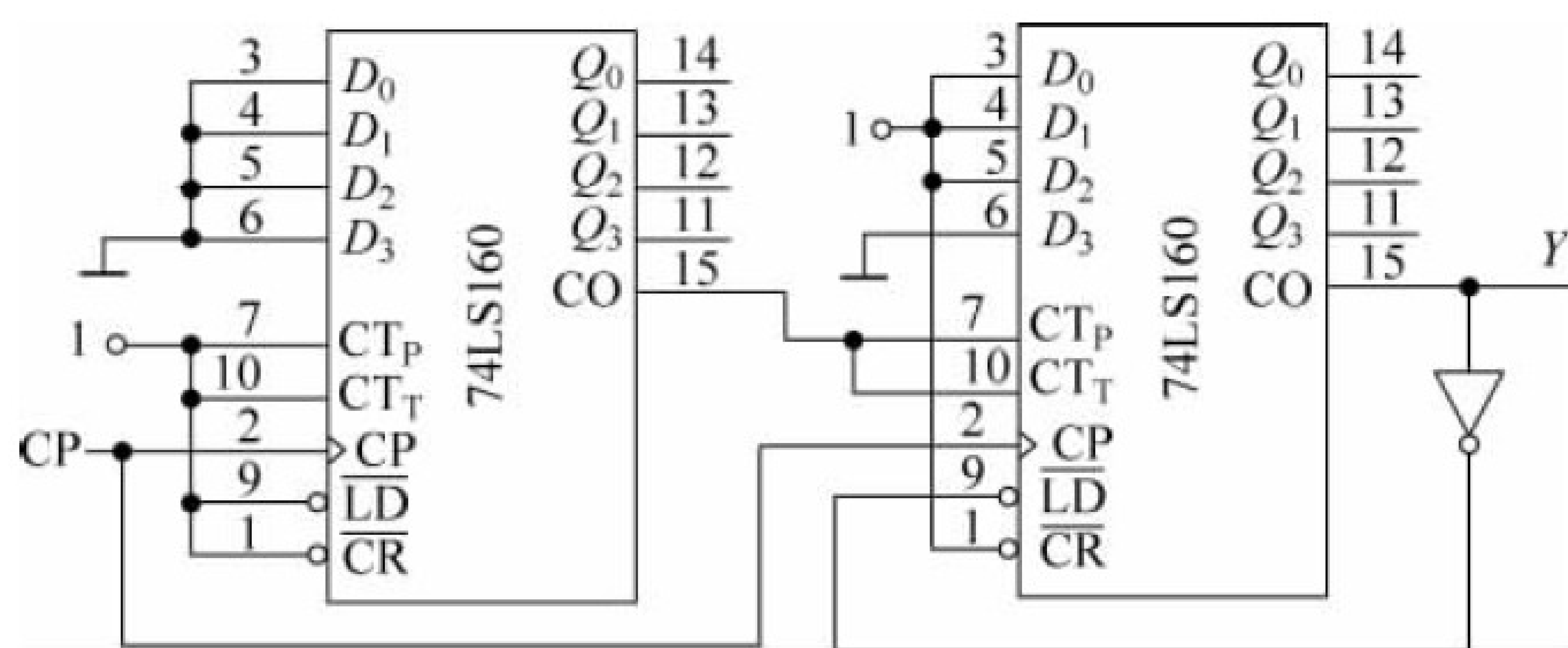
题图 6.21

22. 试分析如题图 6.22 所示电路的逻辑功能。



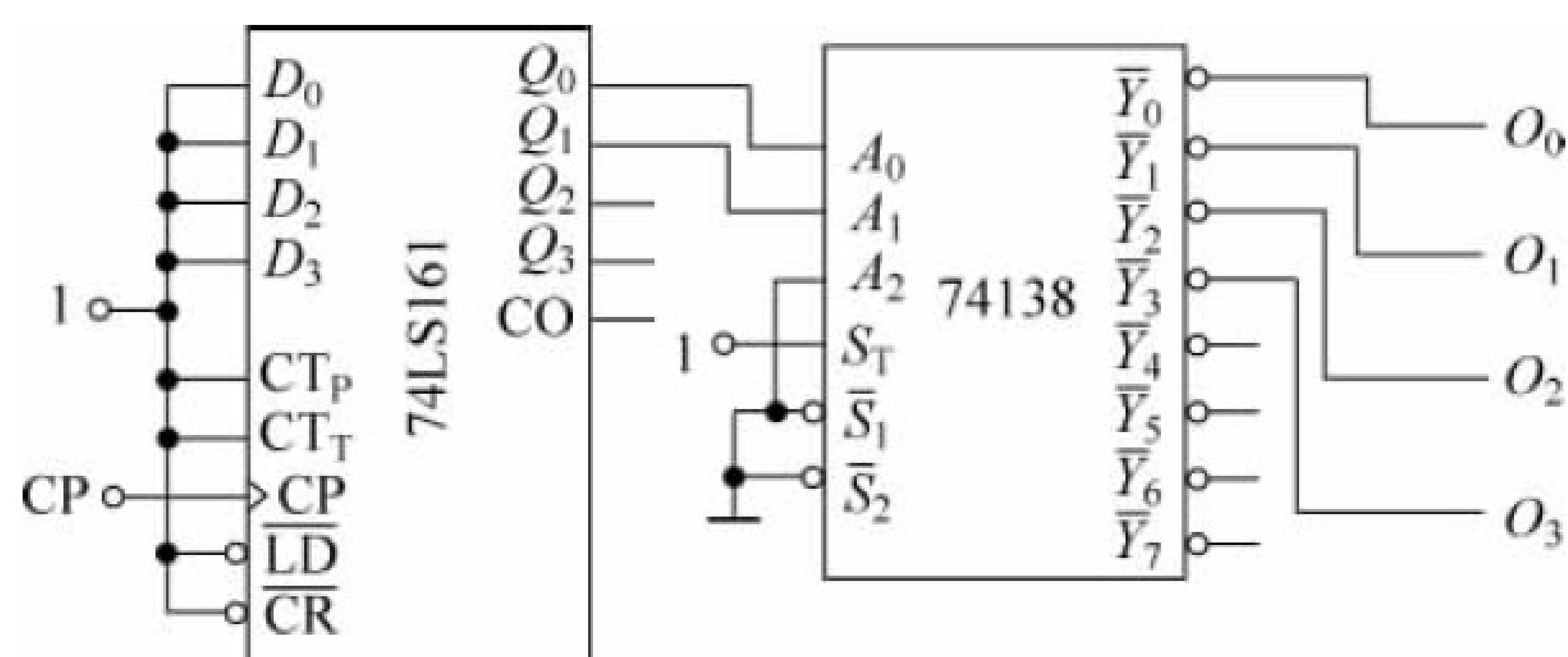
题图 6.22

23. 试分析如题图 6.23 所示电路构成几进制计数器。



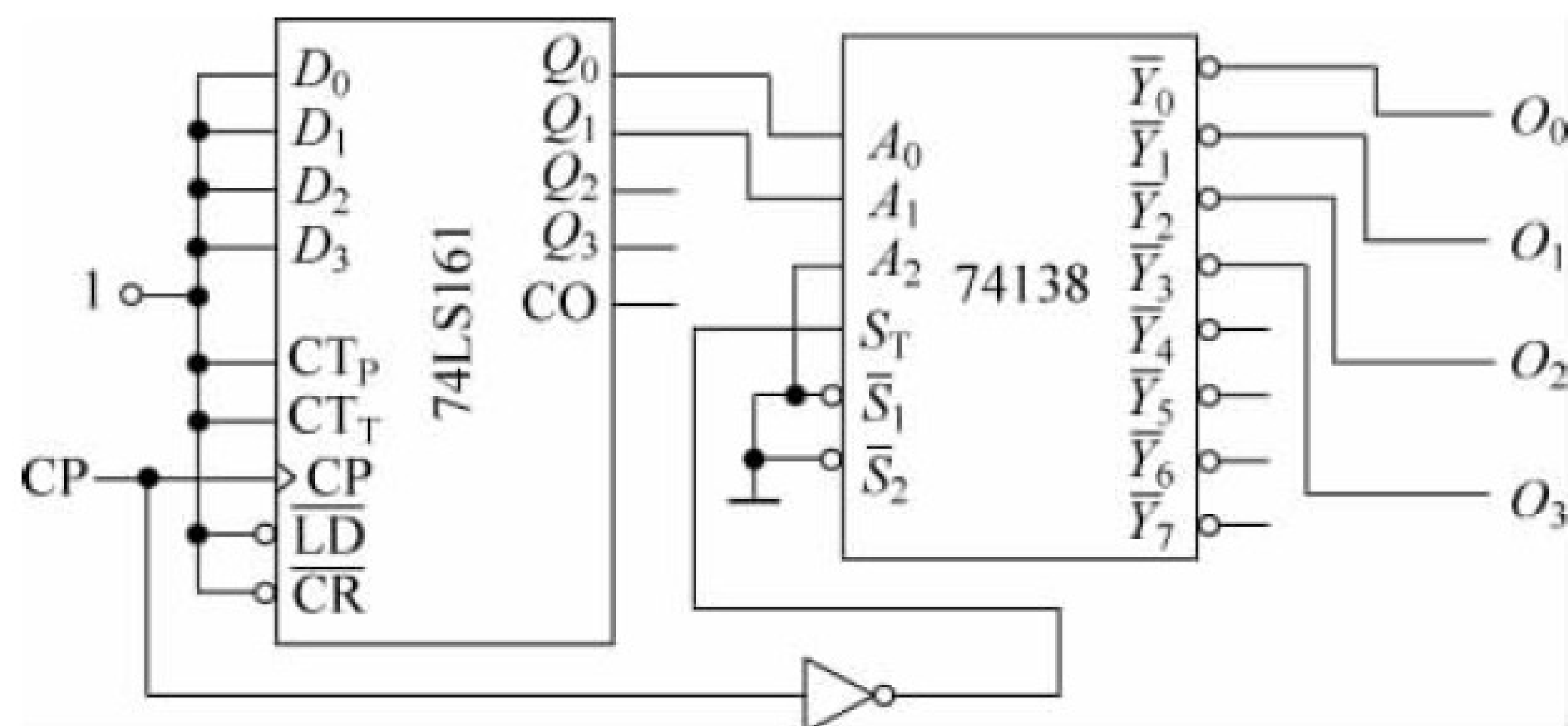
题图 6.23

24. 试分析如题图 6.24 所示电路的逻辑功能。



题图 6.24

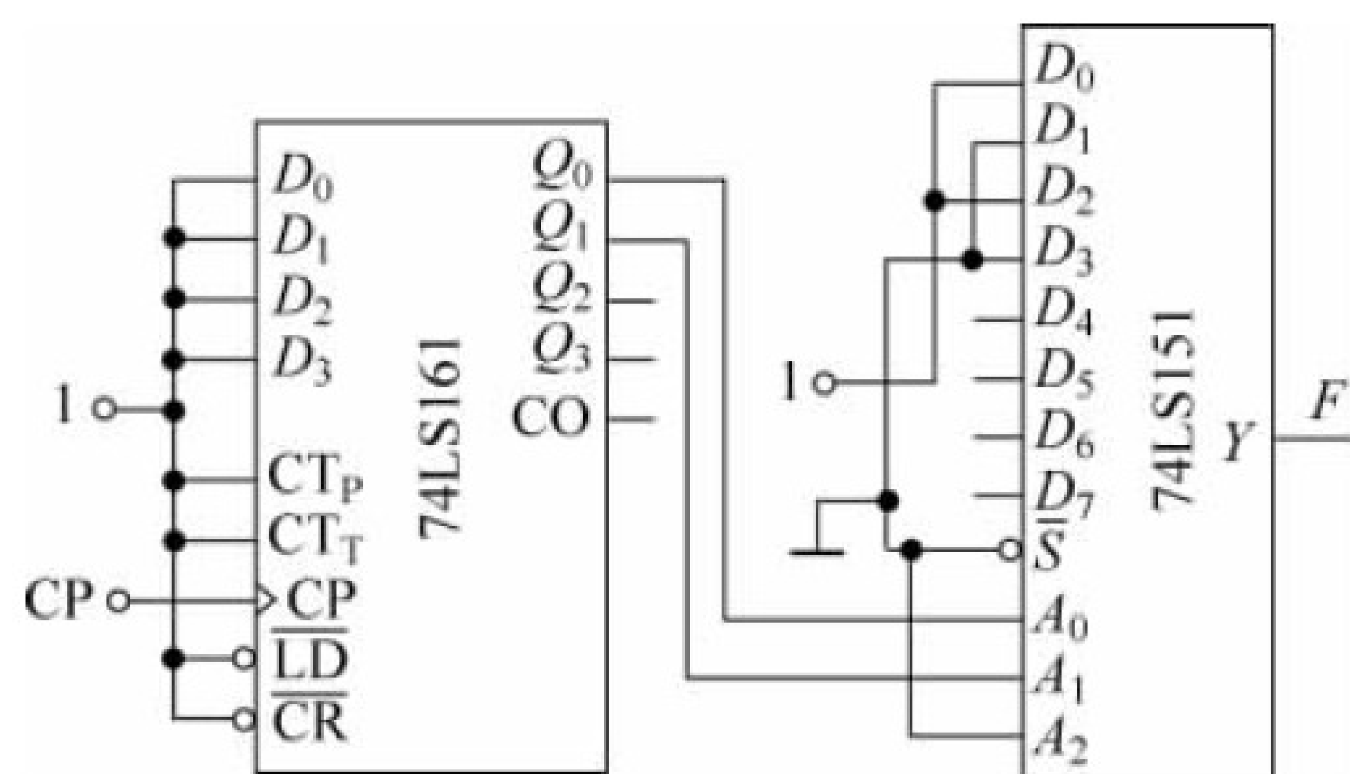
25. 试分析如题图 6.25 所示电路的逻辑功能。



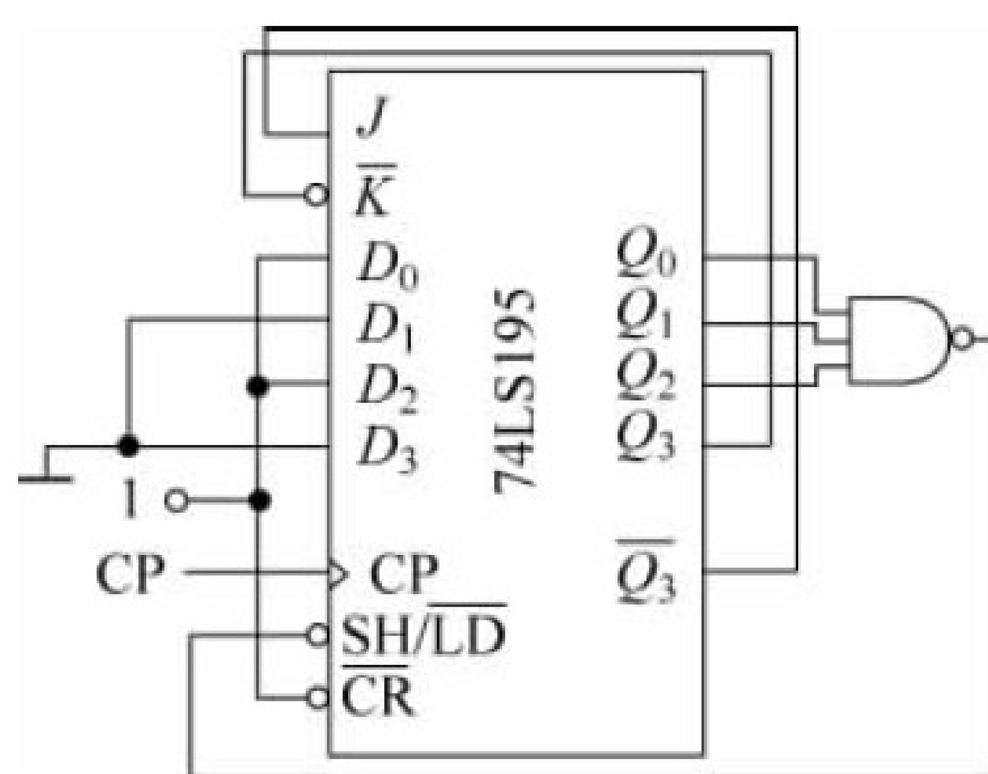
题图 6.25

26. 试分析如题图 6.26 所示电路的逻辑功能。

27. 试分析如题图 6.27 所示电路构成多少进制计数器。

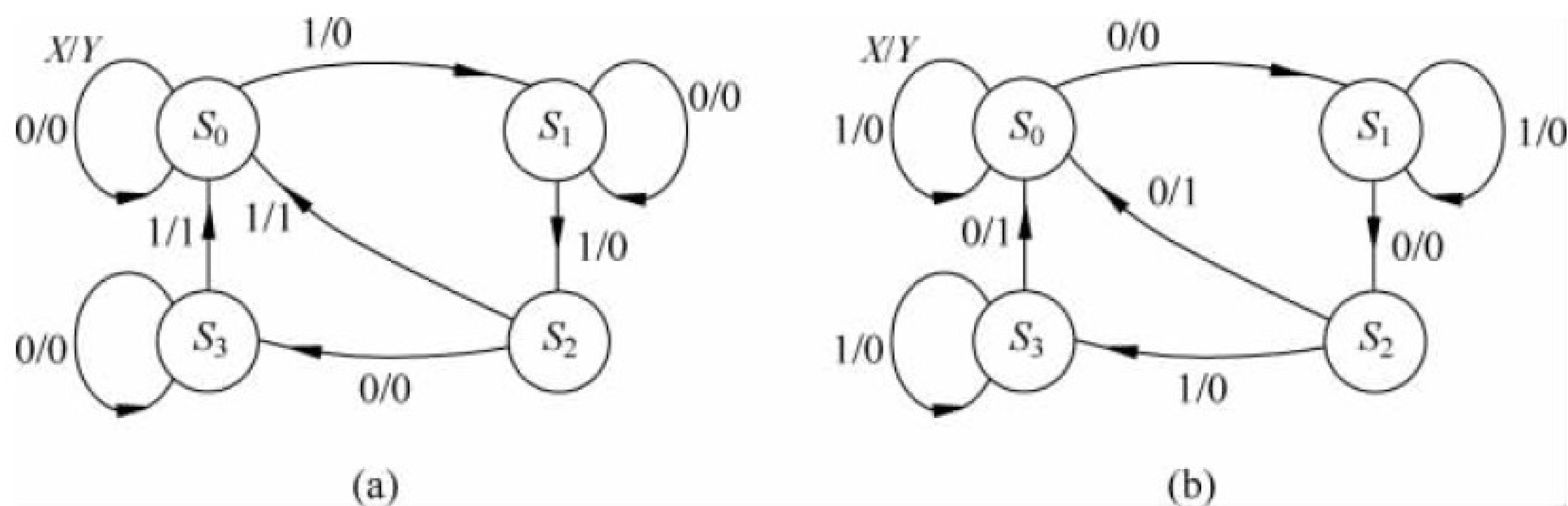


题图 6.26



题图 6.27

28. 试求如题图 6.28 所示各状态图的最简状态图。



题图 6.28

29. 试分析如题图 6.29 所示电路构成几进制计数器。

6-3 应用实践题

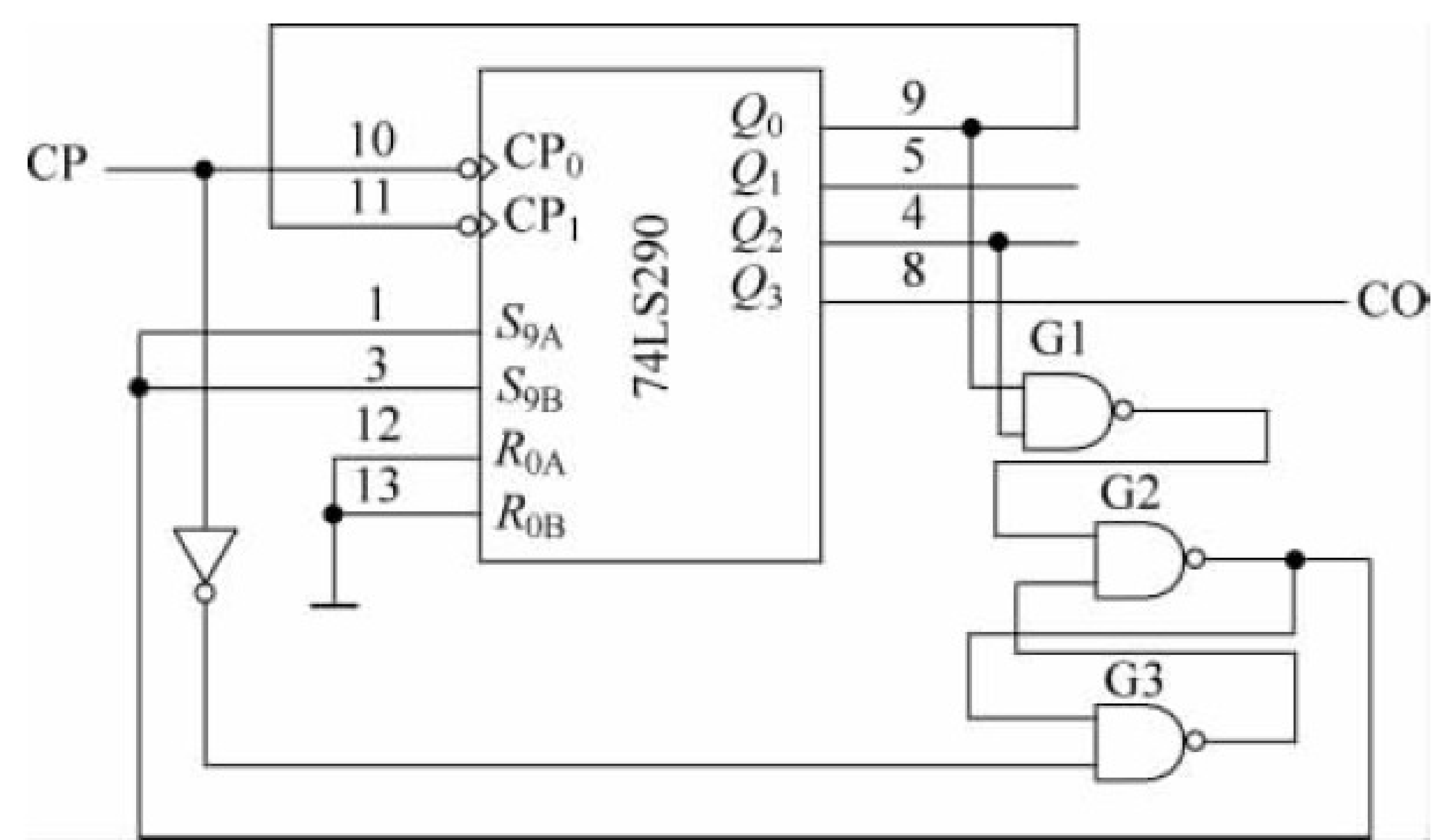
1. 试用 74LS160 的异步清零端设计一个八进制计数器。

2. 试利用 74LS290 的置“9”功能设计一个八进制计数器。

3. 试用 74LS161 的同步置数端设计一个八进制计数器。

4. 试用两片 74LS160 的设计一个十三进制计数器。

5. 试用两片 74LS191 的设计一个二十九进制计数器。



题图　6.29

6. 已知状态转移表如题表 6.1 所示，用 74LS161 设计该同步计数器。

7. 已知状态转移表如题表 6.2 所示，用 74LS191 设计该同步计数器。

题表　6.1

序　号	Q_3　Q_2　Q_1　Q_0
0	0　0　0　0
1	0　0　0　1
2	0　0　1　0
3	0　0　1　1
4	0　1　0　0
5	1　0　1　1
6	1　1　0　0
7	1　1　0　1
8	1　1　1　0
9	1　1　1　1

题表　6.2

序　号	Q_3　Q_2　Q_1　Q_0
0	0　0　0　0
1	0　0　0　1
2	0　1　0　0
3	0　1　0　1
4	0　1　1　0
5	0　1　1　1
6	1　0　0　0
7	1　0　1　1
8	1　1　0　0
9	1　1　0　1
10	1　1　1　0
11	1　1　1　1

8. 试设计一个电动机控制电路。要求该电路有两个控制输入端 X_1 和 X_2，只有在连续两个（或两个以上）时钟脉冲作用期间，两个输入都一致时，电动机才转动。

9. 试用 D 触发器设计一个同步十进制加法计数器。

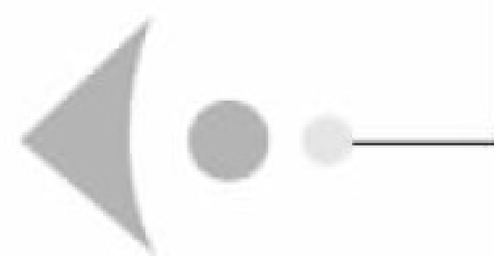
10. 设计一个 6 路输出的顺序正脉冲发生器。

11. 某顺序脉冲发生器具有 4 路输出，前 2 路先顺序输出正脉冲，之后，后 2 路顺序输出负脉冲，设计该顺序脉冲发生器。

12. 设计一个“01110”序列发生器。



第7章



半导体存储器

本章要点：

存储器是计算机系统的五大部件之一。学习本章应理解存储器的概念、种类及其电路构成特点，在此基础上理解随机存储器、只读存储器的原理，芯片实例及其扩展方法，理解利用 EPROM 实现组合逻辑电路的方法。



7.1 概述

谈到存储器，读者不免联想起触发器和寄存器。触发器具有两个稳定状态，可以存储 1 位二进制数。寄存器由触发器组成，n 位并行寄存器可存储 n 位二进制数。触发器、寄存器均具有存储功能，但它们不是存储器。

存储器是计算机的五大部件之一，是用于存储大量数据或信号的半导体器件。前面所讲的触发器和寄存器虽然具有存储的功能，但触发器是小规模集成电路，寄存器为中规模集成电路，采用触发器或寄存器电路结构来存储大量数据是不可能的，也是得不偿失的。因此，存储器应采用专门的电路结构，以满足大量数据存储的要求。

还有一些设备也称为存储器，如硬盘、软盘、光盘等。这些设备具有前面介绍的存储器的存储特点，但它的存储单元不属于电气部件，不可与电气设备直接连接，需要接口电路，因而不是"数字电路""数字电子技术"课程研究的内容，这里所说的存储器是指半导体存储器。

7.1.1 存储器的电路结构及主要参数

可通过图 7.1.1 来理解存储器的电路结构。

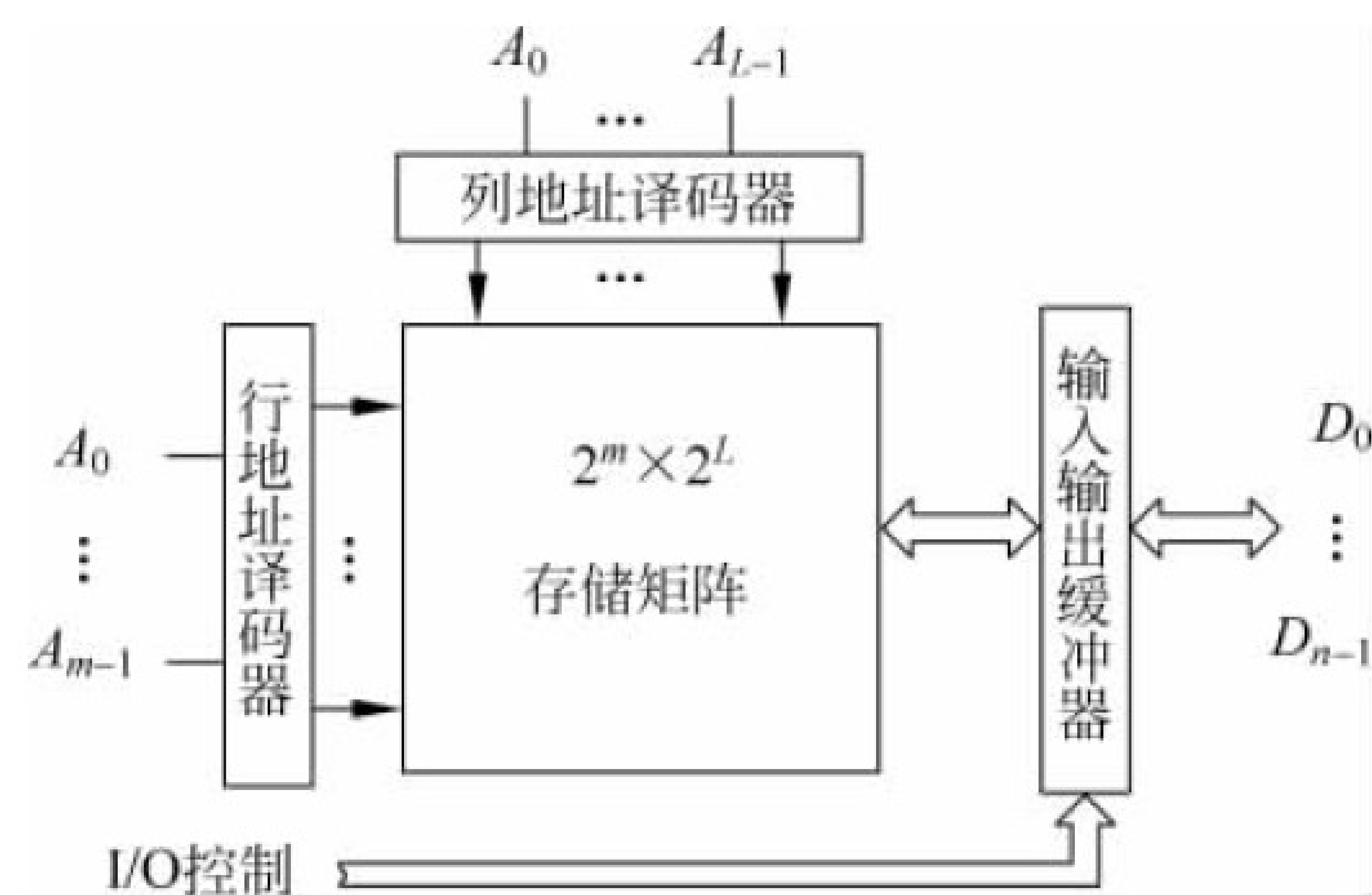


图 7.1.1　存储器结构框图 1

1. 存储矩阵

为满足大量数据存储的要求，存储器采用存储矩阵来存储数据。存储矩阵是由基本存储单元(可存储 n 位二进制数据)构成的存储阵列。存储矩阵所包括的基本存储单元的数目称为存储器的存储容量。

2. 行、列地址译码器

显然，对存储器的一次读只能读出存储器所存储的大量数据中的一个数据。为了能正确读出对应存储单元的存储数据，应通过行、列地址译码器选择对应的存储单元。

当然，也可以不采用行、列地址译码的方式，而用一个地址译码器译出存储器所需要的全部地址线，如图 7.1.2 所示。

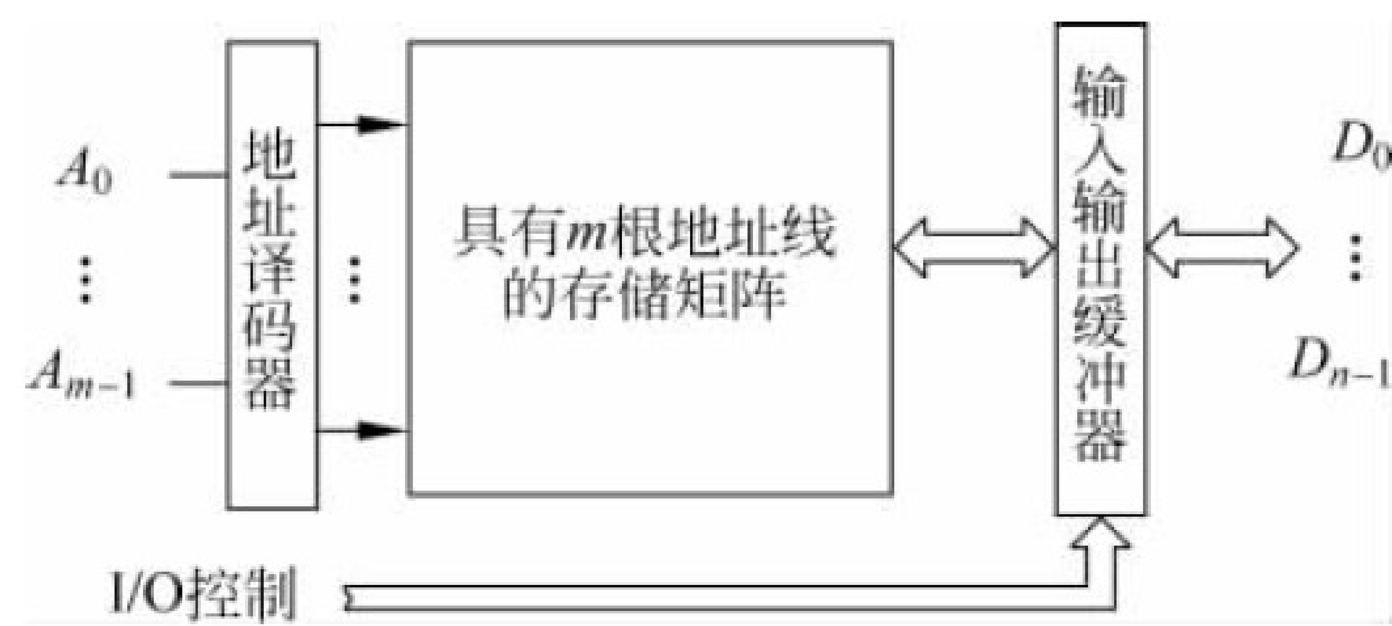


图 7.1.2　存储器结构框图 2

3. 输入输出缓冲及控制

将数据写入存储矩阵的对应存储单元需要一定的时间，从存储矩阵选出对应存储单元的数据也需要一定的时间，为此，可通过输入输出缓冲器及其控制电路完成相应的读写操作。当然，不同类型的存储器，数据读取与写入的方式可能差别非常大，相应的控制电路复杂程度差别也就非常大。

通过上面的分析，可得出存储器两个最基本的参数。

（1）存储器的字数。通常把存储器的每个输出代码称为一个“字”。存储器所具有的地址线的根数 m 反映了存储器的字数。存储器的字数反映了存储器的存储单元的多少。显然，具有 m 根地址线的存储器的字数为 2^m。

（2）存储器的位数。存储器每个输出“字”所具有的二进制位数称为存储器的位数。存储器所具有的数据线的根数 n 反映了存储器的位数。

上面两个参数常用“字数×位数”形式来表示。如 8 位 8KB 存储器表示存储器的字数为 8KB，位数为 8 位，记为“8K×8”。

存储器主要应用于电子计算机及数字系统。现代电子计算机和数字系统处理的数据量均非常大，因此，除上述两个涉及存储器容量的参数外，另一个重要参数就是存取速度。

7.1.2 存储器的种类

存储器是用于存储大量数据的存储设备，高集成度是存储器芯片的基本特点。从集成工艺的角度，存储器可分为 MOS 存储器和双极型存储器两大类。MOS 存储器具有高集成度的优点，逐渐成为大容量存储器的主流，目前微型计算机中的内存均是 MOS 存储器。

从读写方式的角度，存储器可分为只读存储器（Read Only Memory，ROM）和随机存取存储器（Random Access Memory，RAM）。

ROM 从字面意思上看只可读，但现在的 ROM 芯片均可写，按信息的写入方式分为以下几类。

（1）固定 ROM：在工厂制作时就将需要存储的信息用电路结构固定下来，使用时无法再更改。

（2）可编程 ROM（PROM）：由用户按自己的需要写入信息，但只能写入一次，一经写入就不能修改。

（3）可擦可编程 ROM（EPROM）：由用户写入信息后若需要改动时，还可以擦去重写。它具有较大的使用灵活性，但这种改写需使用专门的擦写设备，不可在线改写，而且费时，所以实际使用时常只读不写。

（4）可电改写 ROM（EEPROM）：由用户写入信息后若需要改动时可在线改写，无须专门的擦写设备。它采用以字节为单位的电改写方式，可用于少量的数据改写。

还有一种 ROM 称为快闪存储器（Flash Memory），简称闪存。闪存是 EEPROM 的变种，是一种长寿命、非易失性（在断电情况下仍能保持所存储的数据信息）的存储器，数据删除不是以单个的字节为单位而是以固定的区块为单位。闪存与 EEPROM 不同的是，EEPROM 能在字节水平上进行删除和重写而不是擦写整个芯片，而闪存的大部分芯片需要块擦除。由于闪存采用以数据块为单位的电改写方式，因此具有很高的写入速度。

RAM 可方便地读写数据，但当电源去掉后所存的信息立即消失。目前常用的 RAM 有静态 RAM（Static Random Access Memory，SRAM）、动态 RAM（Dynamic Random

Access Memory,DRAM)等。

静态随机存取存储器 SRAM 是随机存取存储器的一种。所谓的“静态”,是指这种存储器只要保持通电,里面储存的数据就可以保持不变。DRAM 在未断电的情况下,数据也只能暂时保存,需要定期刷新。

顺便说明一下,寄存器虽然不是这里所介绍的存储器,但它依然在计算机中扮演着暂存数据的角色。各种类型的 CPU 中均具有较多的寄存器,在计算机中扮演着不可替代的角色。

复习与思考

7.1.1　ROM 和 RAM 的主要区别是什么?

7.1.2　2KB 的 ROM 具有多少根地址线?

7.1.3　从数据暂存的角度,如何理解寄存器与存储器的相互不可替代性?

7.2　随机存取存储器

随机存取存储器可以随时将数据写入任何指定地址的存储单元中,也可以随时从任一指定的地址的存储单元中,将数据取出(读出)。随机存取存储器的最大优点是读写方便、使用灵活;缺点是存在易失性,即一旦停电,所存储的数据便会丢失。

7.2.1　SRAM 存储原理

各类存储器的存储结构如图 7.1.1 和图 7.1.2 所示,可见,半导体存储器的存储结构大体相同,其数据读写、保存方式主要由存储单元结构决定。

1. 存储单元结构

SRAM 的基本存储单元是在基本 RS 触发器的基础上附加读写控制构成。数据一旦写入,存储单元可维持其原始状态不变,数据读出时不会改变其存储内容。SRAM 的基本存储单元可由 NMOS、CMOS 或双极型晶体管构成。如图 7.2.1 为用 6 只 N 沟通增强型 MOS 管组成的静态存储单元。

它包括如下几个部分:

(1) 用于存储数据的基本 RS 触发器。图中,$T_1 \sim T_4$ 管组成基本 RS 触发器。解释如下:

$Q=1$,T_1 导通,$\bar{Q}=0$,而 $\bar{Q}=0$ 又将使 T_3 截止,确保 $Q=1$,触发器稳定在“1”态。类似地,$Q=0$,$\bar{Q}=1$,触发器稳定在“0”态。

基本 RS 触发器与 T_5、T_6 门控管共同构成 6 管 NMOS 静态基本存储单元。

(2) 读写控制电路。G1、G2、G3、G4,数据线 D,读写控制线 $R/\overline{W}$ 构成读写控制电路。



图 7.2.1　6 管 NMOS 静态存储单元原理图

(3) 存储单元地址选择电路。X_i、T_5、T_6 管构成行地址选择电路。当 $X_i=0$ 时，T_5、T_6 截止，基本 RS 触发器与列线断开；当 $X_i=1$ 时，T_5、T_6 导通，$U_A=\bar{Q}$、$U_B=Q$。

y_i、T_7、T_8 管构成列地址选择电路。当 $y_i=0$ 时，T_7、T_8 截止，存储单元与读写控制电路断开；当 $y_i=1$ 时，T_7、T_8 导通，存储单元与读写控制电路接通。

2. 数据写入过程

(1) 设置数据线 D。将需要写入数据的二进制值赋给数据线 D。

(2) 选中存储单元。令 $x_i=1$，$y_i=1$，则该存储单元被选中，T_5、T_6、T_7、T_8 管均导通。

(3) 令写入控制有效。令 $R/\overline{W}=0$，进行写数操作。由于 $R/\overline{W}=0$，使三态门 G3 禁止，门 G1、G2 工作，所以数据线上的数据经门 G1、G2 写入到存储单元的 Q 和 $\bar{Q}$ 端，$Q=D$。

3. 数据读取过程

(1) 选中存储单元。令 $x_i=1$，$y_i=1$，则该存储单元被选中，T_5、T_6、T_7、T_8 管均导通。

(2) 令读取控制有效。令 $R/\overline{W}=1$，进行读数操作。

由于 $R/\overline{W}=1$，门 G1、G2 禁止，门 G3 工作，所以 $D=Q$。

(3) 外部设备读取数据线上的数据。当然，SRAM 存储单元还可以采用 CMOS、双极型构成，但其存储单元结构大体相同，主要由基本 RS 触发器、存储单元地址选择电路、读写控制电路 3 部分构成，数据读写方法相同，有兴趣的读者可参考相关书籍。

SRAM 性能优良、读写速度快，常用于构成容量要求不大，但速度要求很快的存储系统。计算机中的 Cache(高速缓冲存储器)通常由 SRAM 组成。Cache 又分为 L1Cache(一级缓存)和 L2Cache(二级缓存)，L1Cache 主要是集成在 CPU 内部，而 L2Cache 集成在主板或 CPU 上。

7.2.2 DRAM 电路特点

1. 单管动态 MOS 存储单元特点

DRAM 的存储单元基于 MOS 管栅极电容可以存储电荷的原理设计制作。显然，这样的存储单元结构非常简单，在大容量、高集成度的 RAM 中得到了广泛应用。

早期的存储单元结构采用四管或三管结构。这种结构的优点是控制电路结构简单、读出信号比较强；缺点是电路结构不够简单，不利于提高集成度。

如图 7.2.2 所示为单管动态 MOS 存储单元结构图，由一个 N 沟道增强型 MOS 管和一个电容 C_S组成。

当 $X=1$ 时，T 导通，存储单元被选中，可对存储单元进行读写操作。当执行写操作时，位线上数据经过 T 存入 C_S中。

当进行读操作时，$X=1$，T 导通。假定电容 C_S上储存有正电荷，电压 U_{C_S}为高电平，位线上电压为 0。执行读操作后，电容 C_S经 T 对分布电容 C_D充电，位线电平将上升。

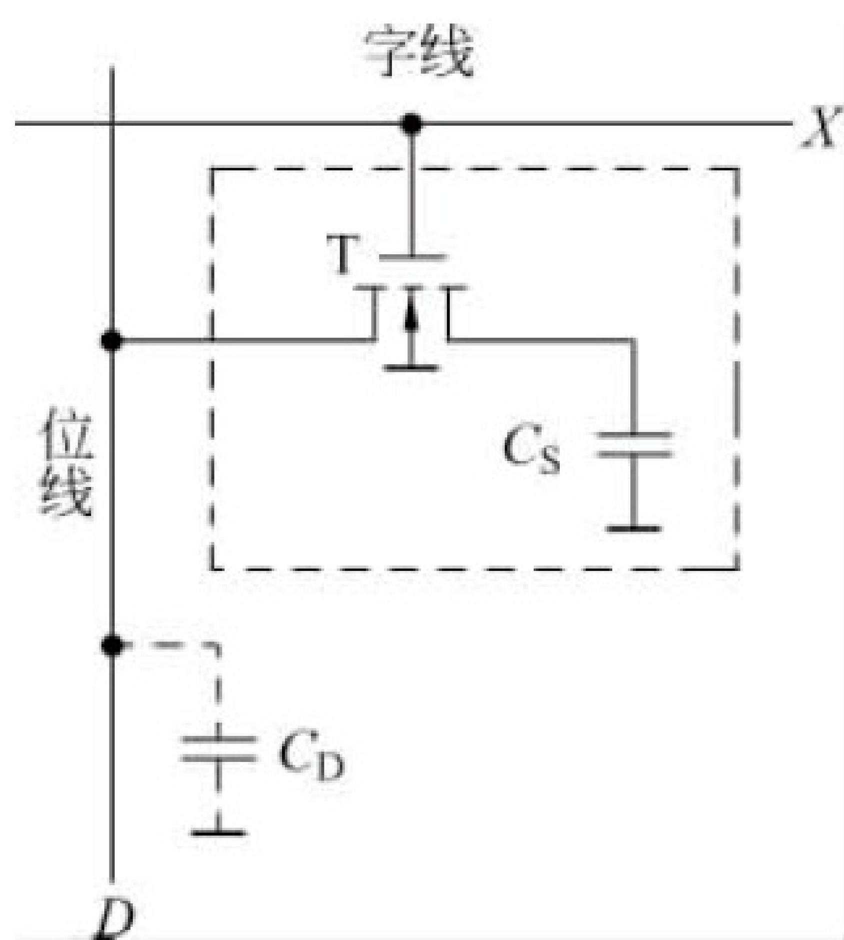


图 7.2.2 单管动态 MOS 存储单元

从制作工艺角度，为节省芯片面积，存储电容 C_S不可能做得很大。所有存储单元均与位线连接，因此，尽管单个 MOS 管栅极分布电容非常小，但总体分布电容 C_D依旧远大于 C_S。

读出时 C_S与 C_D并联，C_D上没有存储电荷，则并联后 C_S上的电荷将向 C_D转移，转移后位线上的电压为 U_{C_D}。但转移前后的总电荷不变，有

$$U_{C_S} \cdot C_S = U_{C_D}(C_D + C_S) \tag{7.2.1}$$

由于 $C_D \gg C_S$，因此，$U_{C_D} \ll U_{C_S}$，读出电压很小。

如读出之前 $U_{C_S}=5\text{V}$，$C_D/C_S=50$，则读出电压 U_{C_D}近似为 0.1V。依照并联的含义，C_S和 C_D上的电压相同，读出之后 U_{C_S}也近似为 0.1V。可见，这种读出是破坏性的读出。

2. DRAM 电路特点

如图 7.2.2 所示单管动态 MOS 存储单元的读出电压只有 0.1V，远远不能满足数字相同高、低电平的要求，需要专门的放大电路对读出信号进行放大。此外，对该单元的读出还是破坏性的读出，需要专门的再生电路恢复被破坏的数据，完成这样功能的电路称为灵敏恢复/读出放大器。此外，恢复后的 U_{C_S}也只能暂时保存高电平状态，需要定期对存储单元进行刷新。当然，灵敏恢复/读出放大器对存储单元的一次读操作可完成对存储单元的刷新，因此，对 DRAM 而言，DRAM 的定期刷新操作事实上是对 DRAM 的定期读操作，只是执行刷新操作时并不真正读数据，刷新时，输出被置成高阻态。

DRAM常用于制作大容量存储器。为适应更大容量存储空间、更多位存储数据的要求，除灵敏恢复/读出放大器外，为了在提高集成度的同时减少器件引脚数量，DRAM电路常使用地址分时输入(也称"地址多路复用)的方式。关于存储器地址的分时输入方式，有兴趣的读者可参考相关书籍。

DRAM外围控制电路非常复杂，包括灵敏恢复/读出放大器、刷新支持、地址分时输入控制等SRAM一般不具备的模块。DRAM数据的修改通过电容充放电来完成，其读写速度也显著地慢于SRAM。此外，DRAM数据只能暂时保存，需要定期刷新才能实现不断电情况下数据的长期保存。可见，与SRAM相比，DRAM在使用上具有诸多不便与不足。

尽管如此，DRAM存储结构极为简单，非常有利于提高集成度，定期刷新操作可得到CPU的良好支持，外围控制电路虽然复杂，但使用上却非常简单，因此，DRAM在大容量存储器方面得到了广泛应用，计算机中的内存条便是典型的DRAM。

计算机中的内存条习惯称为DDR(Double Data Rate，双倍速率)，严格地说DDR应该称为DDR SDRAM(DDR Synchronous Dynamic Random Access Memory，双倍速率同步动态随机存储器)。与传统的单数据速率相比，DDR技术实现了在一个时钟周期内进行两次读/写操作，即在时钟的上升沿和下降沿分别执行一次读/写操作。

7.2.3 RAM芯片实例

RAM应用十分广泛，相应的集成芯片也较多，如图7.2.3所示为静态RAM2114。简要说明如下：

(1) 具有10根地址线和4根数据线。

(2) 片选控制($\overline{CS}$=0芯片工作)。

(3) R/$\overline{W}$为1时读数，为0时写数。

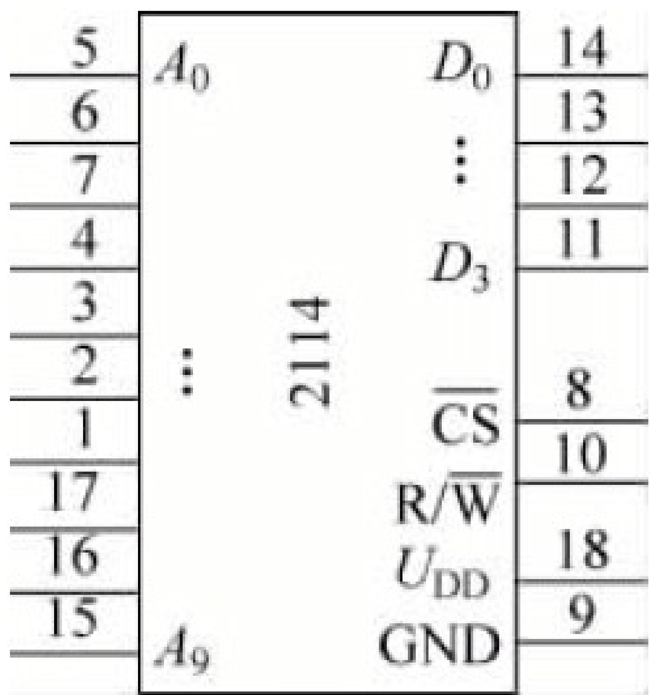


图7.2.3 2114引脚图

复习与思考

7.2.1 参照图7.2.1设计一个2行、2列具有4个存储单元的4×1的SRAM。

7.2.2 动态MOS电路的工作特点是什么?

7.2.3 总结SRAM、DRAM各自的特点。

7.3 只读存储器

RAM的最大缺点是当电源去掉后所存的信号立即消失，而在数字系统中有不少信息，如数学函数、固定程序、常用字符等，需要长期保存。为此，需要将这些信息固定存入断电后信息不会消失的存储器。这类存储器在正常工作时一般只读不写，称为只读存储器(ROM)。

7.3.1 固定 ROM

固定 ROM 的存储矩阵在生产时便已设计好,以后不能修改,常采用二极管来实现固定 ROM 的存储矩阵,电路结构和如图 7.1.2 所示的存储器结构相同。如图 7.3.1 所示为具有 2 位地址输入和 4 位数据输出的 ROM 电路,包括地址译码器、存储矩阵、输入输出缓冲及控制 3 个部分。

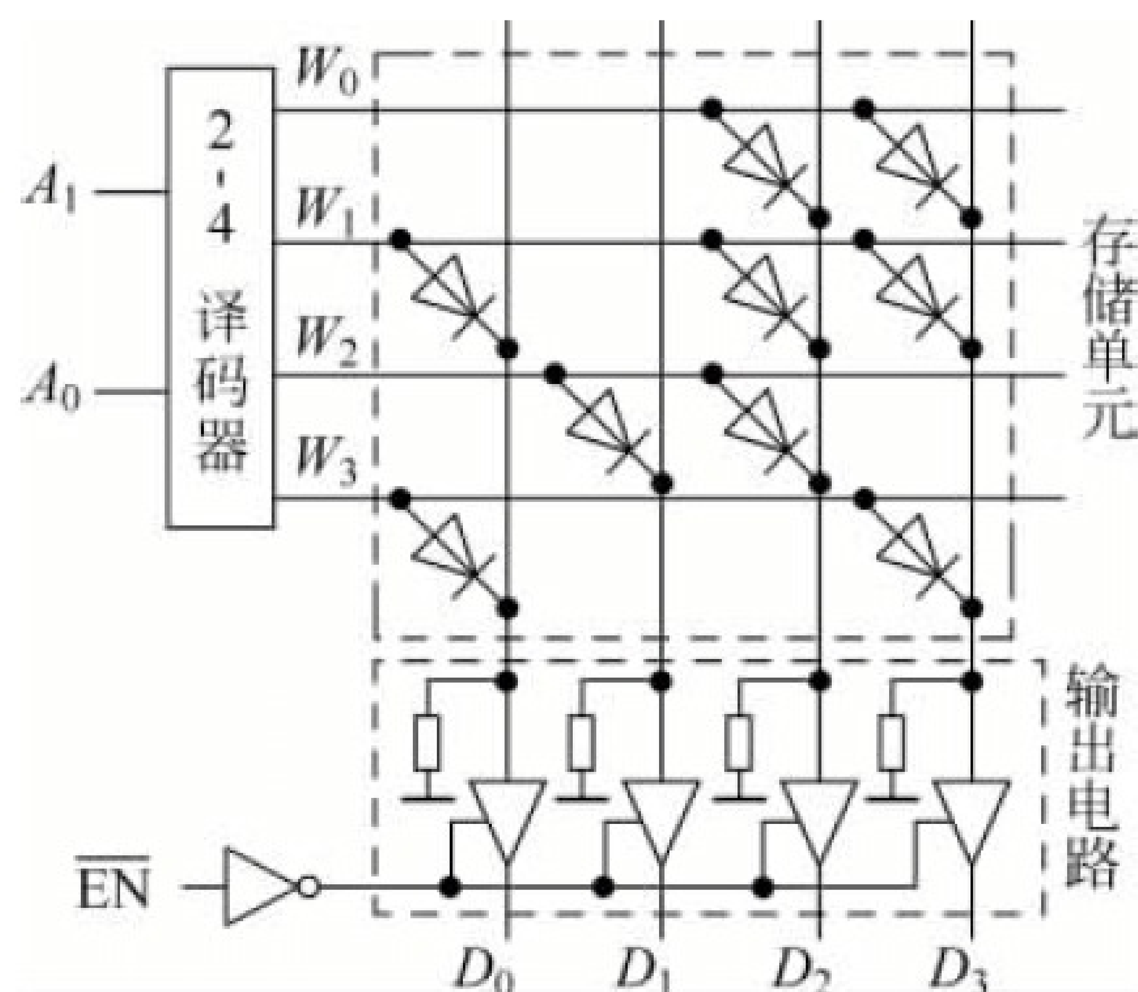


图 7.3.1 用二极管组成的 4×4 ROM 电路

它的存储单元全部使用二极管,地址译码器是一个以高电平为有效输出电平的 2-4 全译码电路。译码器输出的 4 根线 W_0～W_3 称为字线,存储矩阵输出的 4 根线 D_0～D_3 为位线。

不难看出,字线(W)和位线(D)的每个交叉点都是一个存储单元。交叉点处接有二极管时相当于存“1”,没有接二极管时相当于存“0”。交叉点的数目就是存储单元数。如图 7.3.1 所示 ROM 的存储量为 4×4。

当读数据时,只要输入指定的地址码,并令 $\overline{EN}=0$,则指定地址内各存储单元所存的数据经三态输出门出现在输出数据线上。如输入地址(A_1A_0)为 01 时,令 $\overline{EN}=0$,那么输出数据线上的数据($D_3D_2D_1D_0$)为 1101。类似 A_1A_0 为 11 时,$D_3D_2D_1D_0$ 为 1001。

如图 7.3.1 所示固定 ROM 的最大优点是电路结构极为简单,数据读取速度非常快,长期保存性能极好;缺点是数据不能修改,只适应于数据永久不修改的应用场合。

7.3.2 ROM 的写入

在很多场合,要求 ROM 具有写的功能,但 ROM 的写的功能将在一定程度上损伤 ROM 对数据的长期保存性能。对 ROM 执行写的方式主要有专用设备写入、在线写入等方式。

1. 专用设备写入

专用设备写入为 PROM、EPROM 采用的写入模式,数据的长期保存性能极好,常用于数据极少修改的应用场合。

当然,ROM 的数据写入方式是由其存储单元结构决定的,如图 7.3.2 所示为 PROM 存储单元原理图,如图 7.3.3 所示为 EPROM 存储单元原理图。

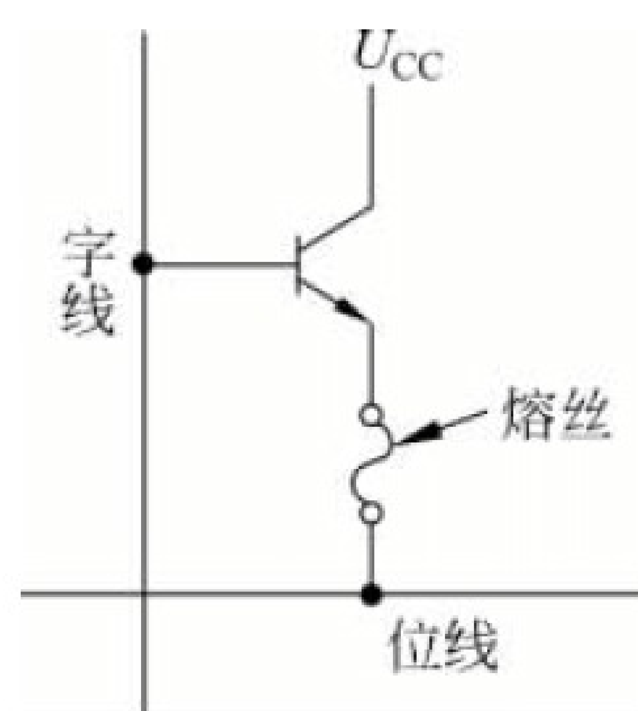


图 7.3.2 PROM 存储单元原理

如图 7.3.2 所示,可一次写入的存储单元的存储器件为三极管,为 PROM 采用的存储结构。接有三极管的存储单

元为 1。出厂时所有单元均接有三极管，对应的熔丝也是通的，因此，出厂时所有单元相当于全部存入了 1。显然，对该存储单元的正常读不会烧断熔丝，可利用专门的 PROM 写入板，使三极管流过足以烧断熔丝的脉冲电流，断开三极管与存储单元的连接，完成数据“0”的写入。当然，熔丝烧断后无法再连上，PROM 中的数据只能写入一次。

类似地，PROM 也可采用接有三极管的存储单元为 0 的存储方法实现，有兴趣的读者可参考如图 7.3.2 所示原理图设计该 ROM 的存储单元。

如图 7.3.3 所示可擦除可写入的存储单元的存储器件为叠栅注入 MOS 管（Stacked Gate Injection MOS，SIMOS），为 EPROM 采用的存储结构。与字线相连的栅极 G 为控制栅。中间的栅极 G_f埋在绝缘层，为浮置栅。SIMOS 管结构如图 7.3.4 所示，为 N 沟道增强型 MOS 管。

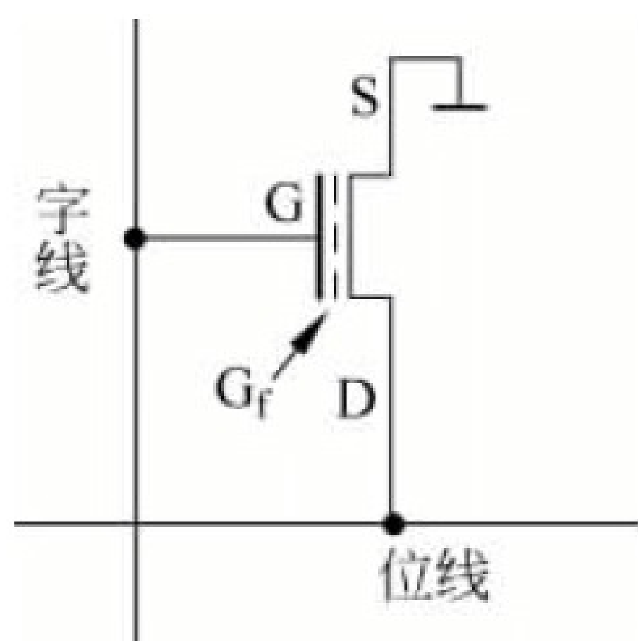


图 7.3.3　EPROM 存储单元原理

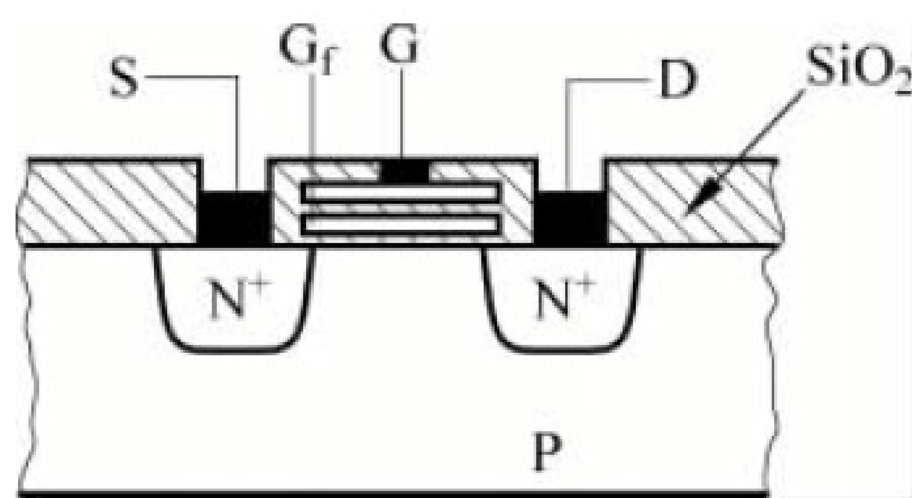


图 7.3.4　SIMOS 结构

如果没有浮置栅，或者浮置栅上没有负电荷，则在控制栅 G 上加正常高电平时，D-S 间将形成导电通路，如图 7.3.3 所示存储单元数据为 0。当浮置栅充有足够的负电荷时，正常逻辑电平下，浮置栅上的负电场将抵消控制栅上的正电场，D-S 间无导电通路，如图 7.3.3 所示存储单元数据为 1。

可见，当存储单元中的浮置栅被注入了负电荷时，相当于写入 1，未注入电荷的单元相当于存入 0。可通过专门的 EPROM 写入板给需要写 1 的单元注入负电荷从而完成对 EPROM 的写入。

由于浮置栅被埋在绝缘层中，它所俘获的电子很难被泄漏掉，故可长期保存。此外，某存储单元被写入 1 后，浮置栅中的负电荷将被长期保存，无法修改为 0，需要使用专门的设备为浮置栅提供泄电通道，擦除原始数据，之后才可再次写入。

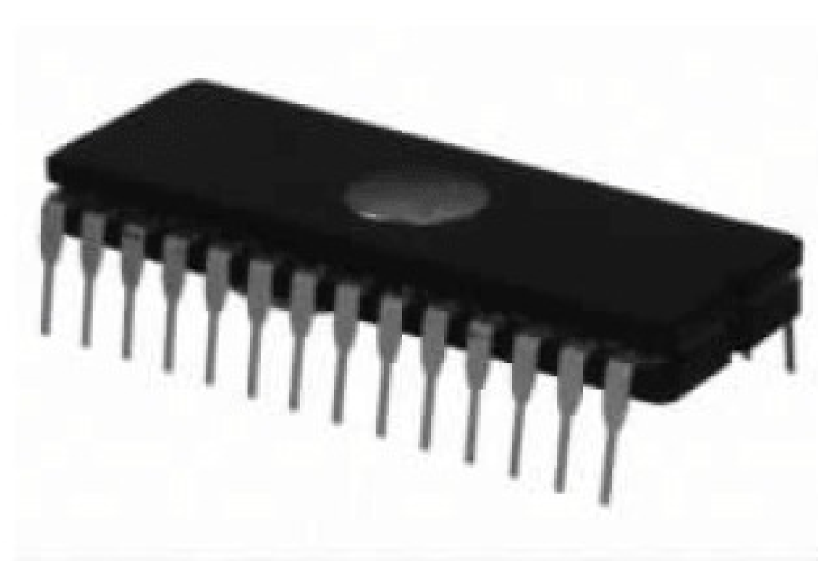


图 7.3.5　EPROM 芯片外观

可使用紫外线照射仪对 SIMOS 的栅极氧化层进行长时间照射，在绝缘层中提供一个缓慢的泄电通道，实现浮置栅中储存的负电荷中的放电，从而完成原始数据的擦除。为方便实现数据的擦除，集成 EPROM 芯片均提供了一个石英玻璃窗口用于紫外线照射，EPROM 芯片外观如图 7.3.5 所示。通过该石英玻璃窗口，紫外线照射仪可在 20～30min 完成数据的擦除，阳光下大约需要 1 周完成擦除，一般的荧光灯可能

需要照射数年时间才可完成擦除。

当然，为提高 ROM 芯片的数据长期保存性能，正常情况下，可用不干胶布密封贴住石英玻璃窗口，防止自然光长期照射擦除数据。

2. 在线写入

采用专用设备写入的 ROM 芯片数据写入非常不方便，写入时间也很长，不适合于需要经常修改的 ROM 应用。

如图 7.3.6 所示为在线写入 ROM 存储单元结构示意图，存储单元的存储器件 T_1 为浮栅隧道氧化层(Floating Gate Tunneling Oxide，Flotox)MOS 管，为 EEPROM 采用的存储结构。Flotox 管的结构图如图 7.3.7 所示。

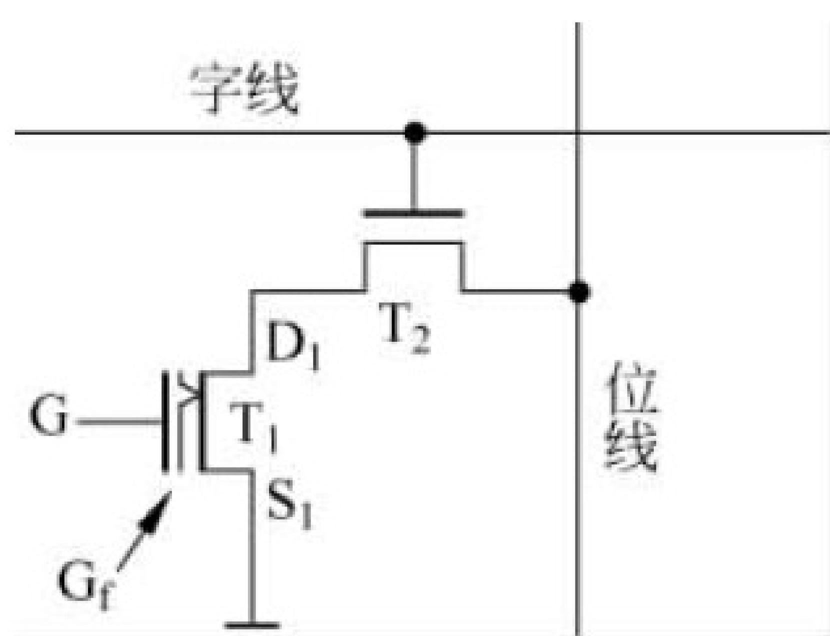


图 7.3.6　EEPROM 存储单元

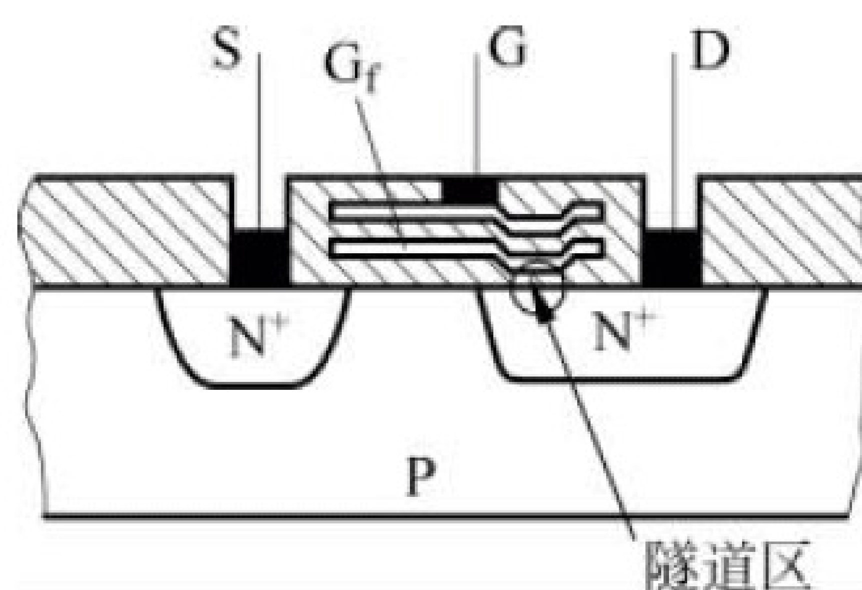


图 7.3.7　Flotox 管的结构

如图 7.3.6 所示存储单元中，字线加有效电平，存储单元被选中，T_2管 D-S 间形成导电通路。当 Flotox 管的浮置栅没有负电荷时，若在控制栅 G 上加正常高电平，则 T_1的 D-S 间形成导电通路，如图 7.3.6 所示存储单元数据为 0。当浮置栅充有足够的负电荷时，正常逻辑电平下，浮置栅上的负电场将抵消控制栅上的正电场，T_1的 D-S 间无导电通路，如图 7.3.6 所示存储单元数据为 1。可见，当存储单元中的存储器件 T_1浮置栅被注入了负电荷时，相当于写入 1，未注入电荷的单元相当于存入 0。

SIMOS 管需要先用紫外线照射仪擦除先前写入的数据，之后才可写入数据，数据的写入、擦除均需要专门的设备，数据长期保存性能好，但数据改写不方便。Flotox 管浮置栅与漏极之间有一个氧化层极薄(厚度一般在 20nm 以下)的区域，这个区域称为隧道区。当隧道区的电场达到一定强度时，漏极 D 和浮置栅之间将出现导电通道，电子可以双向通过，形成电流，这种现象称为隧道效应。由于存在隧道效应，所以可在线擦除存储单元数据，无须专门的设备对芯片进行擦除与写入。

为了提高擦除与写入的可靠性，保护隧道区的超薄氧化层，如图 7.3.6 所示存储单元中增加了选通管 T_2。字线加有效电平，存储单元被选中，T_2管 D-S 间将形成导电通路。若 T_1的浮置栅没有负电荷，T_1、T_2均导通，位线与地接通，存储单元数据为 0。若 T_1的浮置栅有足够的负电荷，T_1、T_2均不导通，存储单元数据为 1。

尽管 EEPROM 改用电压信号对存储单元内容进行擦除与写入，但其擦写时间为 ms 级，因此，数字系统正常工作时，EEPROM 依然主要工作在只读状态。

为了提高写入速度，20 世纪 80 年代末，快闪存储器(Flash Memory)问世，如图 7.3.8 所示为快闪存储器存储单元结构示意图，存储单元的存储器件为叠栅结构的 MOS 管。

快闪存储器的存储原理和前面的 EPROM、EEPROM 并无本质区别，当存储单元中叠栅结构 MOS 管的浮置栅有足够的负电荷时，存储单元数据为 1；当浮置栅没有负电荷时，存储单元数据为 0。和 EEPROM 相比，快闪存储器的存储单元去掉了选通管 T_2，有利于进一步提高集成度。

当然，快闪存储器存储单元中的叠栅结构的 MOS 管并非 EPROM 中的 SIMOS 管，只是结构相似而已，其结构如图 7.3.9 所示。SIMOS 管中浮置栅和氧化层的厚度一般为30～40nm，快闪存储器中的叠栅管只有 10～15nm，而且，浮栅与源区的重叠部分是由源区的横向扩散形成，面积极小，形成了一个隧道区，可利用隧道效应在线擦除存储单元数据，而无须专门的设备对芯片进行擦除与写入。

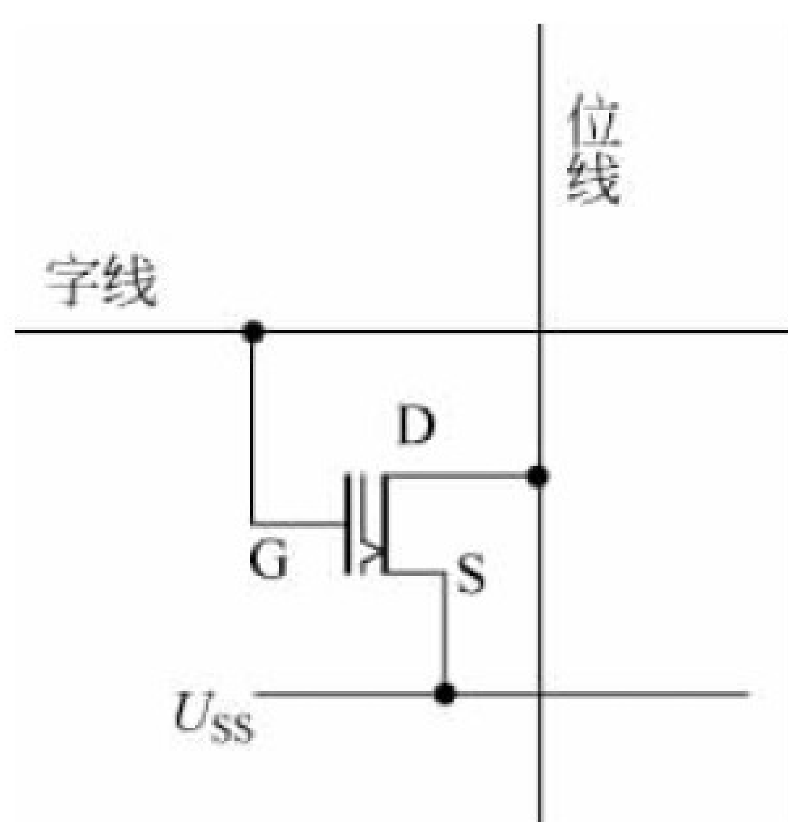


图 7.3.8 快闪存储器存储单元

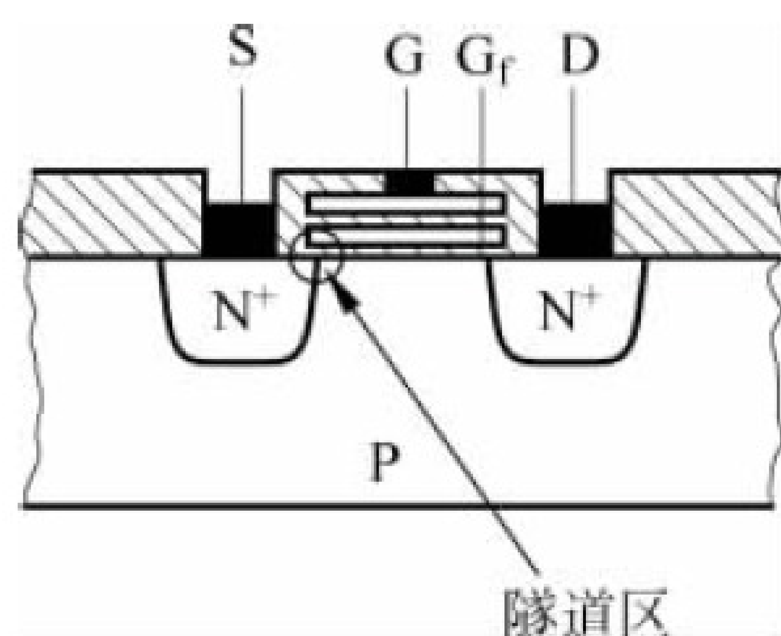


图 7.3.9 快闪存储器的叠栅管

快闪存储器的存储单元中的源极一般都是连接在一起的。因此，当在源极加高电平擦除存储单元数据时，所有字线为 0 的存储单元将同时被擦除。利用这个特点，尽管单个单元擦除时间依然不快，但可以将存储器分成几个区，一次将区内的所有存储单元同时擦除，大大提高了擦除速度，快闪的含义也源于此。



7.3.3 ROM 集成芯片实例

在数字系统中，经常使用的 ROM 类型有 EPROM、EEPROM。参考 EPROM 芯片如 2716(2K×8)、2732(4K×8)、2764(8K × 8)、27128 (16K × 8)、27256 (32K × 8) 等。参考 EEPROM 芯片如 2816(2K×8)、2817(2K×8)等。

下面以 2716 为例介绍集成 ROM 芯片的应用特点。

图 7.3.10 为 2716 的引脚图，简要说明如下：

(1) 具有 11 根地址线 A_{10}～A_0、8 根数据线 D_7～D_0。

(2) 输出使能控制端(当 $\overline{OE}=0$ 时，存储单元内容允许输出)。

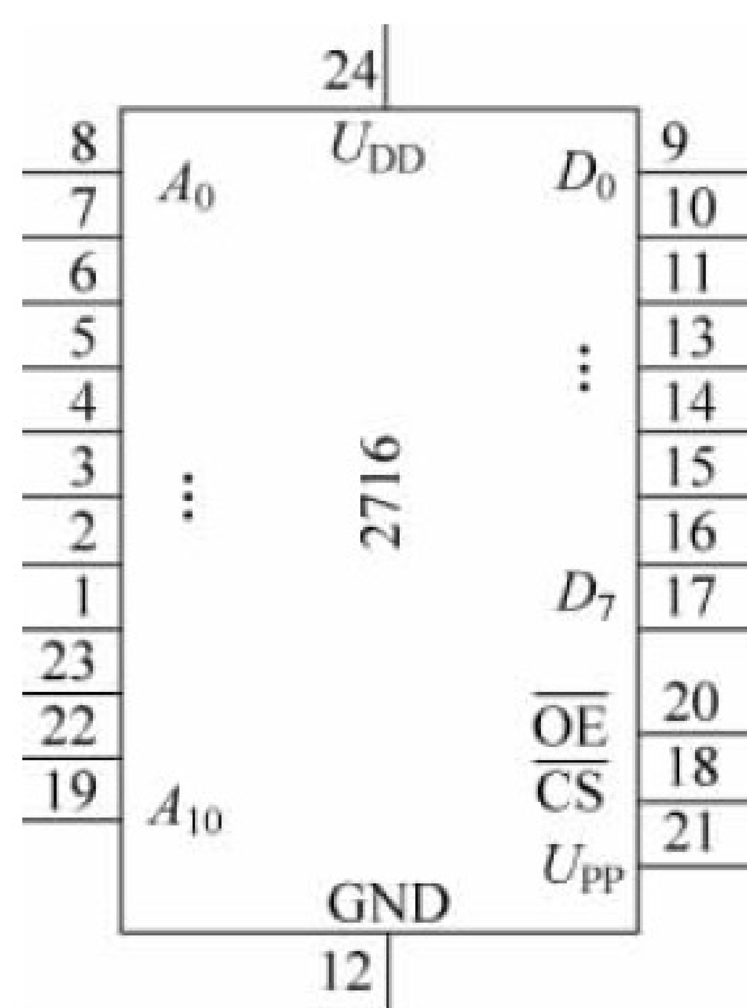


图 7.3.10 2716 芯片

(3) 片选控制(当$\overline{CS}=0$时,芯片工作)。

(4) 专用设备擦除、专用设备写入。

读2716比较简单,可令$\overline{OE}=\overline{CS}=0$,将要读出的存储单元地址$A_{10}\sim A_0$送上地址线,经过数百纳秒后,对应存储单元的数据便出现在数据线上。

将数据写入到2716的方法如下:

(1) 先用擦除设备将芯片中数据擦除,所有单元内容为"全0"。

(2) 将U_{PP}接25V,令$\overline{OE}=1$,将要写入的存储单元地址及数据送上地址线及数据线,待地址及数据稳定后在$\overline{CS}$端加一个50ms的正脉冲即可完成一个单元的写入。

复习与思考

7.3.1 能否去掉如图7.3.6所示EEPROM存储单元中的T2管并实现数据存储?若能去掉,应如何连接电路?

7.3.2 快闪存储器中的"快闪"是指"快闪擦除"还是"快闪写入"?

7.4 存储器的扩展



当使用一片存储器器件不能满足对存储容量的要求时,可以将若干片存储器组合起来,连接成一个容量更大的存储器。

存储器扩展主要有字扩展、位扩展两种基本方式。

当每片存储器中的字数不够用时,可以采用字扩展法,将多片组合成字数更多的存储器。当每片RAM中的字数已经够用,而每个字的位数不够用时,可以采用位扩展法,将多片RAM组合成位数更多的RAM存储器。

下面以RAM为例介绍存储器的扩展方法。

【例7.4.1】 用2114实现一个1K×16的存储系统。

解 (1) 分析。2114具有10根地址线,字数为1K,正好满足要求;2114的位数为4,题中要求位数为16,故需要4片2114。

(2) 定义数据位。令片1~片4的数据位分别对应$D_0\sim D_{15}$。

(3) 连接地址线及控制线。扩展后的系统对存储器的单次读写均来自各芯片相同序号的存储单元,可将片1~片4的地址输入线及控制输入线短接。

(4) 画出电路。用2114实现的1K×16的存储系统电路如图7.4.1所示,图中粗线表示总线。

【例7.4.2】 用2114实现一个2K×4的存储系统。

解 (1) 分析。2114的位数为4,正好满足要求;2114具有10根地址线,字数为1K,题中要求字数为2K,故需要两片2114。

(2) 定义数据位。数据位不扩展,片1、片2的数据位相对应短接。

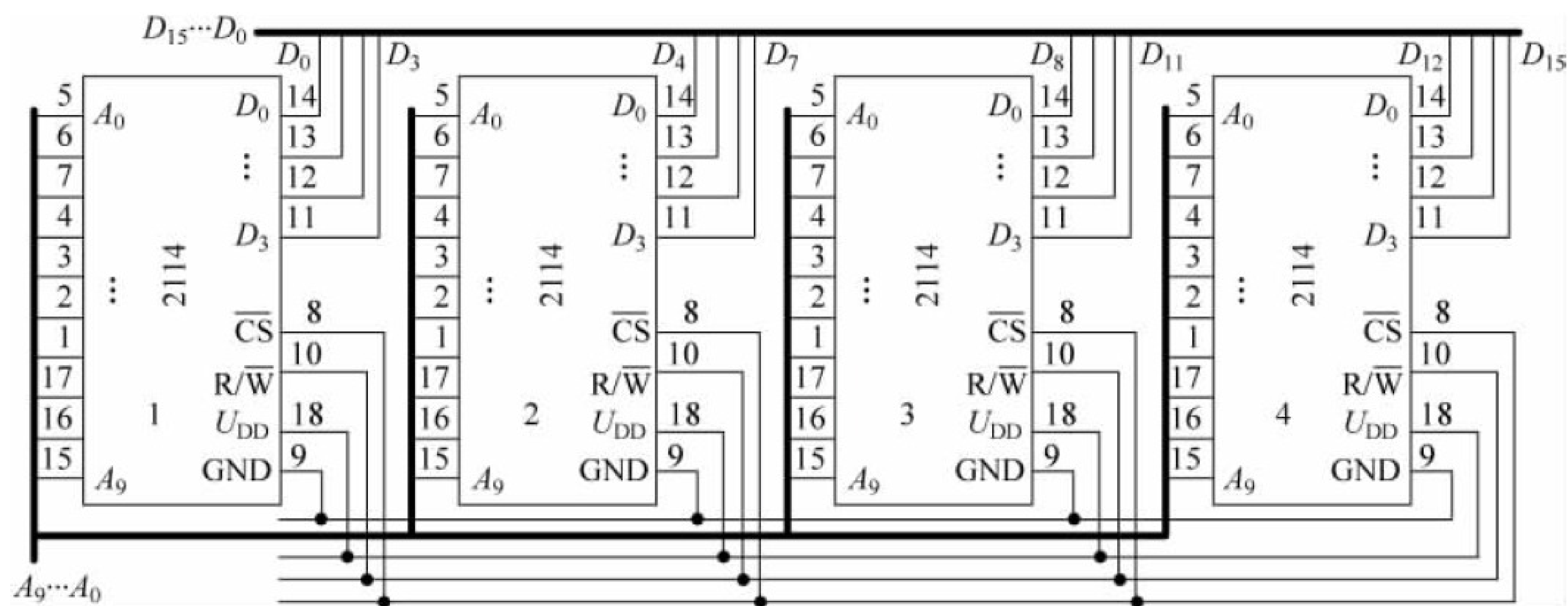


图 7.4.1　例 7.4.1 的图

注意：工程实践中，输入可以短接，输出一般不可短接。ROM 的数据位当然是输出位，一般不可以短接。多片具有三态输出特性的芯片，在满足任何时刻最多只有一路输出有效的情况下可以短接。

(3) 给片 1、片 2 分配不同的地址存储空间。设片 1 的地址存储空间为 0～1K－1，片 2 的地址存储空间为 1K～2K－1，则片 1、片 2 整体的存储空间为 0～2K－1，实现了 1 个 2K×4 的存储系统。

(4) 地址存储空间的实现。芯片的存储空间由地址输入线决定。2K 的存储系统具有 11 根地址线，依照(3)中的分配方法，片 1、片 2 的地址空间分配表如表 7.4.1 所示。

表 7.4.1　例 7.4.2 的地址空间分配表

地　址　线		A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
片 1	最小	0	0	0	0	0	0	0	0	0	0	0
	最大	0	1	1	1	1	1	1	1	1	1	1
片 2	最小	1	0	0	0	0	0	0	0	0	0	0
	最大	1	1	1	1	1	1	1	1	1	1	1

可见，当 $A_{10}=0$ 时，片 1 工作。当 $A_{10}=1$ 时，片 2 工作。令第 11 根地址线 A_{10} 接片 1 的$\overline{CS}$，A_{10} 取反后接片 2 的$\overline{CS}$。

(5) 画出电路。电路如图 7.4.2 所示。

【例 7.4.3】　分析图 7.4.3 所示存储系统的地址空间。

解　片 1、片 2 的$\overline{CS}$相互连接构成一个 1K×8 的存储系统，类似片 3、片 4。A_{10}、A_{11} 加到 2-4 译码器的输入端，译码器的输出$\bar{y}_1$ 用作片 1、片 2 的片选信号；$\bar{y}_3$ 用作片 3、片 4 的片选信号。

当 $A_{11}A_{10}=01$ 时，片 1、片 2 工作，其地址空间为 010000000000～011111111111，即 1K～2K－1。

当 $A_{11}A_{10}=11$ 时，片 3、片 4 工作，其地址空间为 110000000000～111111111111，即

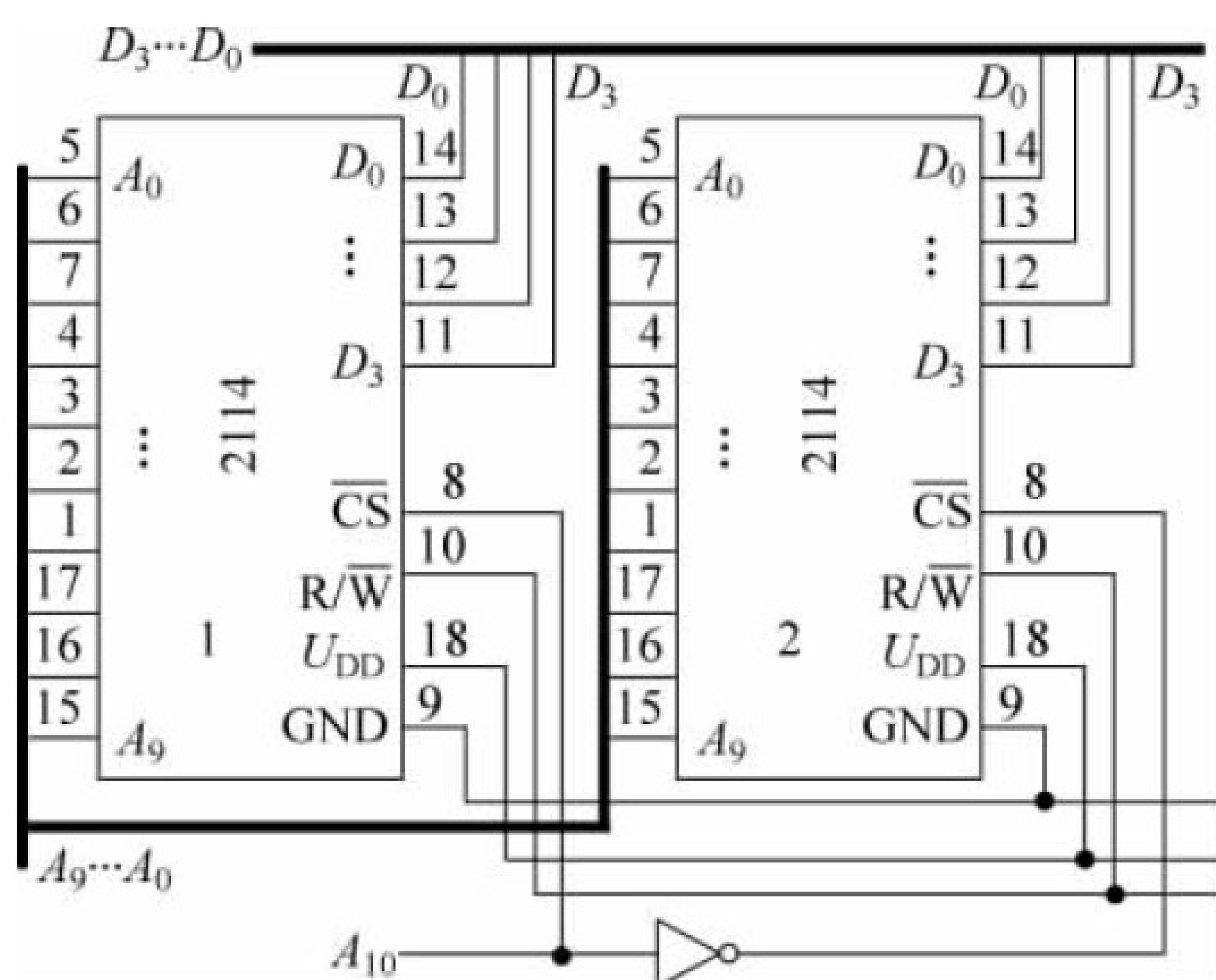


图 7.4.2 例 7.4.2 的图

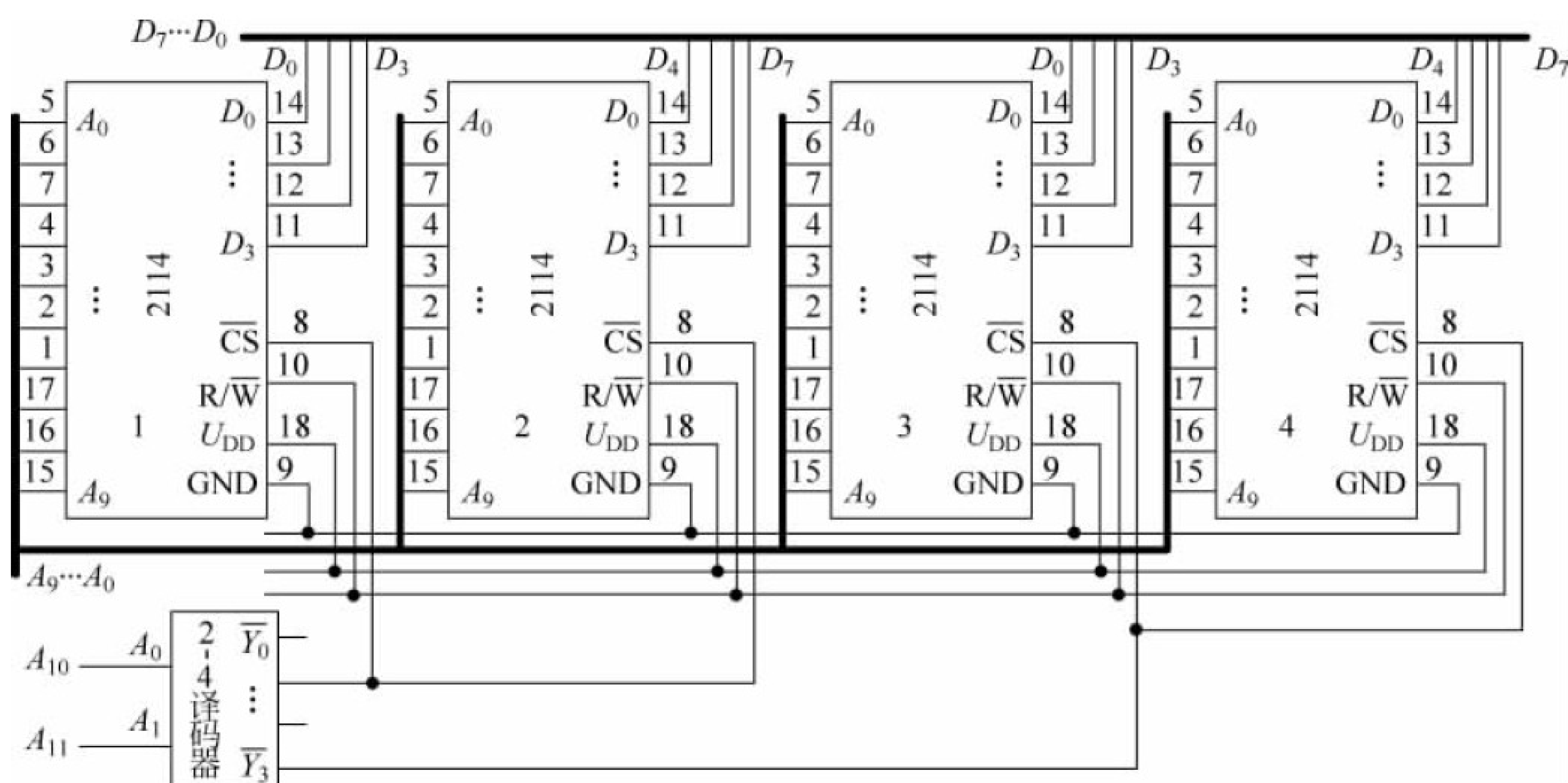


图 7.4.3 例 7.4.3 的图

3K～4K－1。

顺便说明一点，RAM 的扩展方法同样适合 ROM。

如图 7.4.4 所示电路为用两片 2716 实现的 4K×8 的 ROM 存储系统。

2716 为(2K×8)的 ROM 芯片，图 7.4.4 中，片 1、片 2 的数据位相对应短接，数据位未进行扩展。

2716 具有 11 根地址线 A_{10}～A_0。图 7.4.4 中，A_{11} 接片 1 的$\overline{CS}$，当 A_{11}＝0 时，片 1 工作。片 1 的地址存储空间为 0～2K－1。

A_{11} 取反后接片 2 的$\overline{CS}$，当 A_{11}＝1 时，片 2 工作。片 2 的地址存储空间为 2K～4K－1。

因此，片 1、片 2 整体的存储空间为 0～4K－1，实现了一个 4K×8 的存储系统。

复习与思考

7.4.1　分析如图 7.4.5 所示存储系统的存储地址空间及存储容量大小。

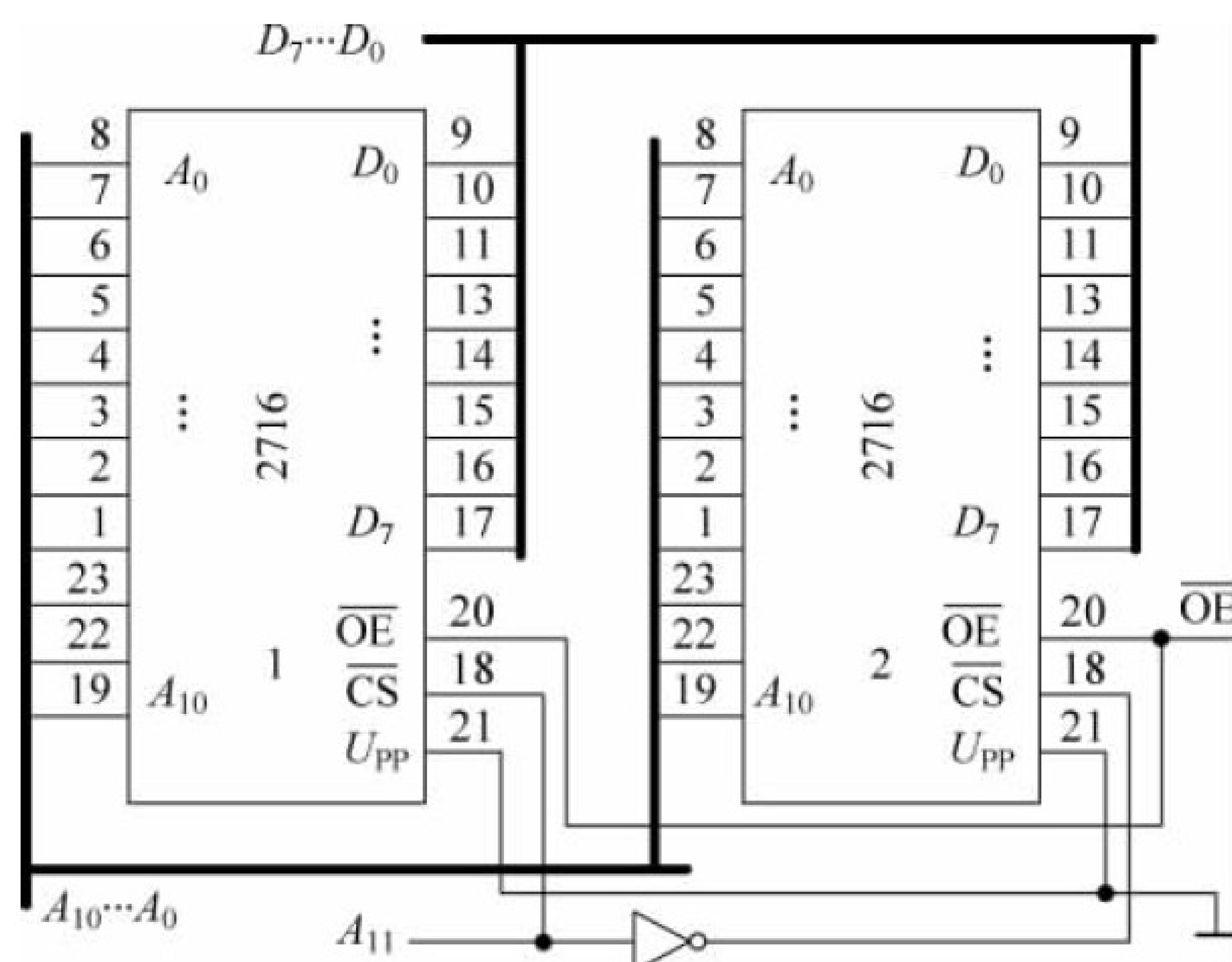


图 7.4.4 用两片 2716 实现的 4K×8 的 ROM 系统

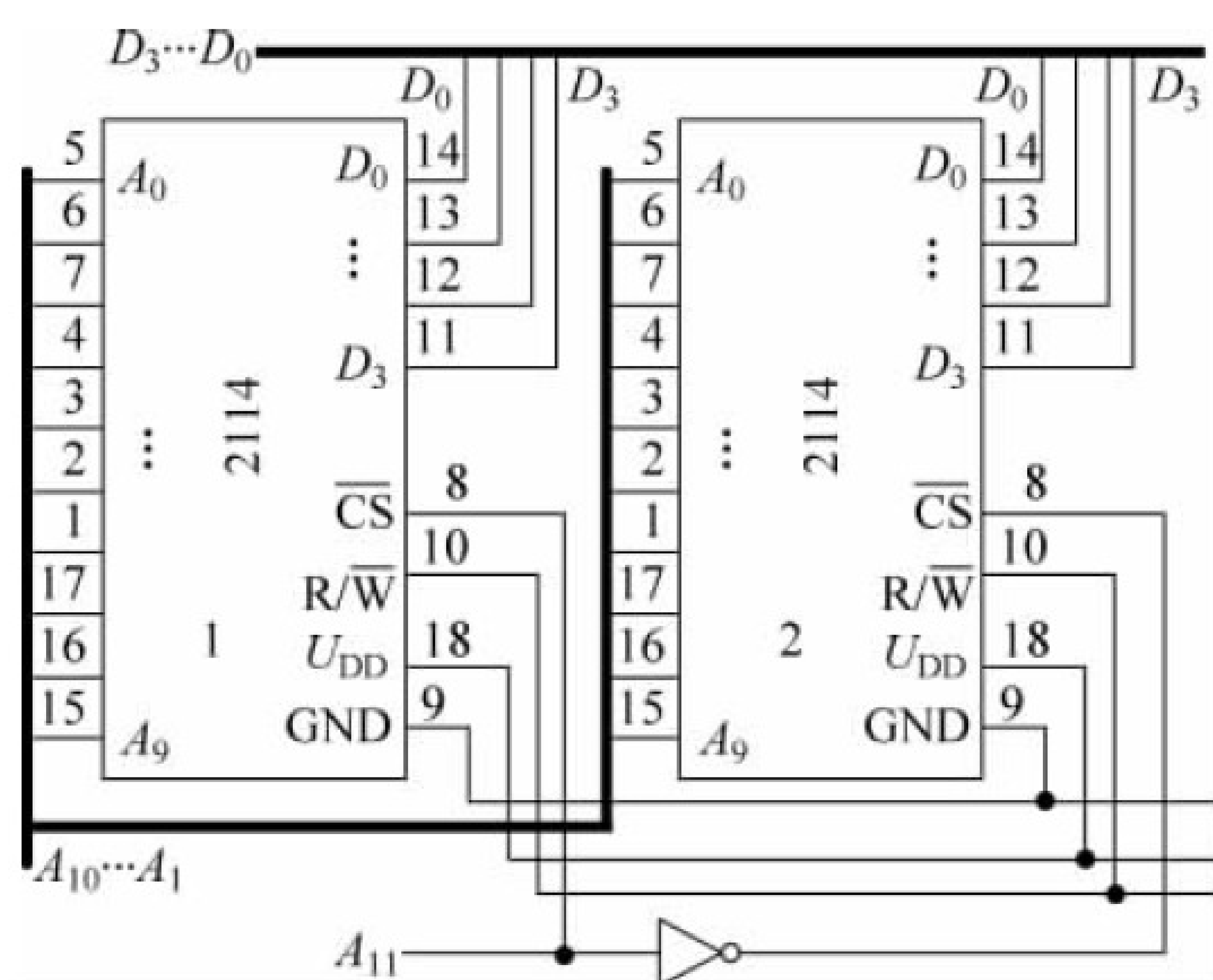


图 7.4.5 复习与思考 7.4.1 的图

7.4.2 分析如图 7.4.6 所示存储系统的存储地址空间及存储容量大小。

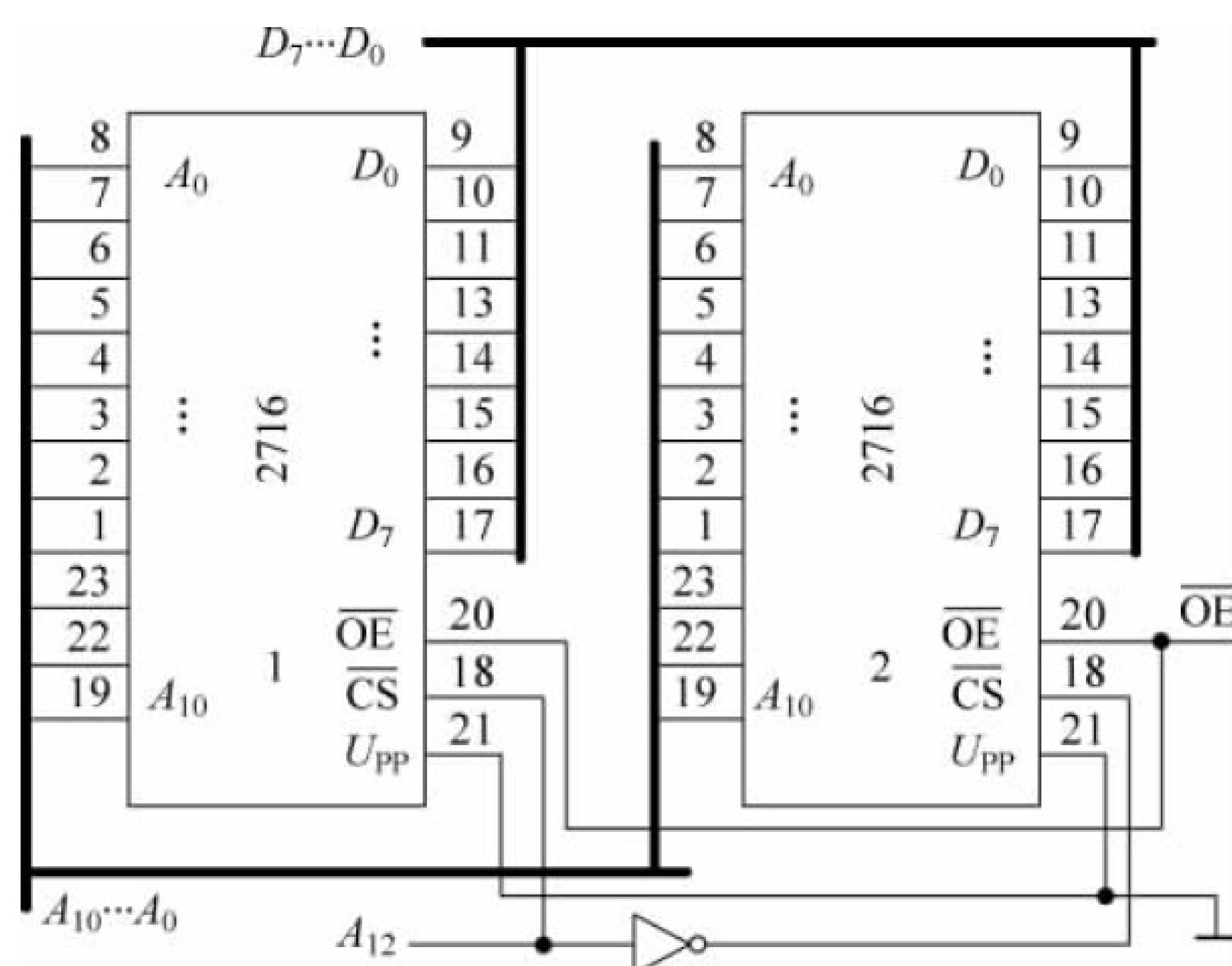


图 7.4.6 复习与思考 7.4.2 的图

7.5　用 ROM 实现组合逻辑电路



ROM 除作为存储器使用外，还可以用来产生任意函数，设计任意组合逻辑电路。这便是前面提到的用大规模集成逻辑器件设计组合逻辑电路的方法。

利用 ROM 可实现任意组合逻辑电路的原因是，ROM 本身就是组合逻辑电路。例如，具有 4 个存储空间的 ROM 的 0～3 个单元的 4 个数据分别为 3、5、7、9，当前读出数据为 7 的原因是当前输入的地址单元编号为 2。可见，ROM 的输出只与当前的输入有关，是典型的组合逻辑电路。

可从另一个角度理解如图 7.3.1 所示的固定 ROM。

令输出使能控制端$\overline{\mathrm{EN}}=0$，A_1A_0为输入，$D_0\sim D_3$为输出，可得出如表 7.5.1 所示的真值表。

由表 7.5.1 可以写出

$$\begin{cases} D_3=\overline{A}_1\overline{A}_0+\overline{A}_1A_0+A_1A_0 \\ D_2=\overline{A}_1\overline{A}_0+\overline{A}_1A_0+A_1\overline{A}_0 \\ D_1=A_1\overline{A}_0 \\ D_0=\overline{A}_1A_0+A_1A_0 \end{cases} \tag{7.5.1}$$

可见，如图 7.3.1 所示固定 ROM 为一个 4 输出组合逻辑电路。

表 7.5.1　真值表

A_1	A_0	D_3	D_2	D_1	D_0
0	0	1	1	0	0
0	1	1	1	0	1
1	0	0	1	1	0
1	1	1	0	0	1

观察图 7.3.1、表 7.5.1 和式(7.5.1)，如果将地址端A_1、A_0作为输入变量，将D_3、D_2、D_1、D_0作为输出变量，适当地选择存储单元内容，则D_3、D_2、D_1、D_0可为任意的两变量逻辑函数。

可通过下面的例子来理解适当地选择如图 7.3.1 所示 4×4 的 ROM 存储单元内容可实现任意两变量的逻辑函数。

【例 7.5.1】　用 4×4 ROM 阵列实现下面的函数。

$$Y_3=A\oplus B,\quad Y_2=\overline{A}\overline{B},\quad Y_1=A+B,\quad Y_0=AB$$

解　(1) 作上述函数的真值表，如表 7.5.2 所示。

(2) 设置输入输出。4×4 ROM 有 2 个地址输入端A_1、A_0和 4 个数据输出端$D_3\sim D_0$，可令$A=A_1$，$B=A_0$，$Y_3=D_3$，$Y_2=D_2$，$Y_1=D_1$，$Y_0=D_0$。则表 7.5.2 可用如表 7.5.3 所示的真值表来表示。

(3) 编写程序。由表 7.5.3，4×4 ROM 的 4 个单元内容分别为“0100”“1010”“1010”“0011”。

(4) 最终电路。用二极管实现上面 4 个函数的 4×4 ROM 电路，如图 7.5.1 所示。

表 7.5.2 真值表 1

A	B	Y_3	Y_2	Y_1	Y_0
0	0	0	1	0	0
0	1	1	0	1	0
1	0	1	0	1	0
1	1	0	0	1	1

表 7.5.3 真值表 2

A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	1	0	0
0	1	1	0	1	0
1	0	1	0	1	0
1	1	0	0	1	1

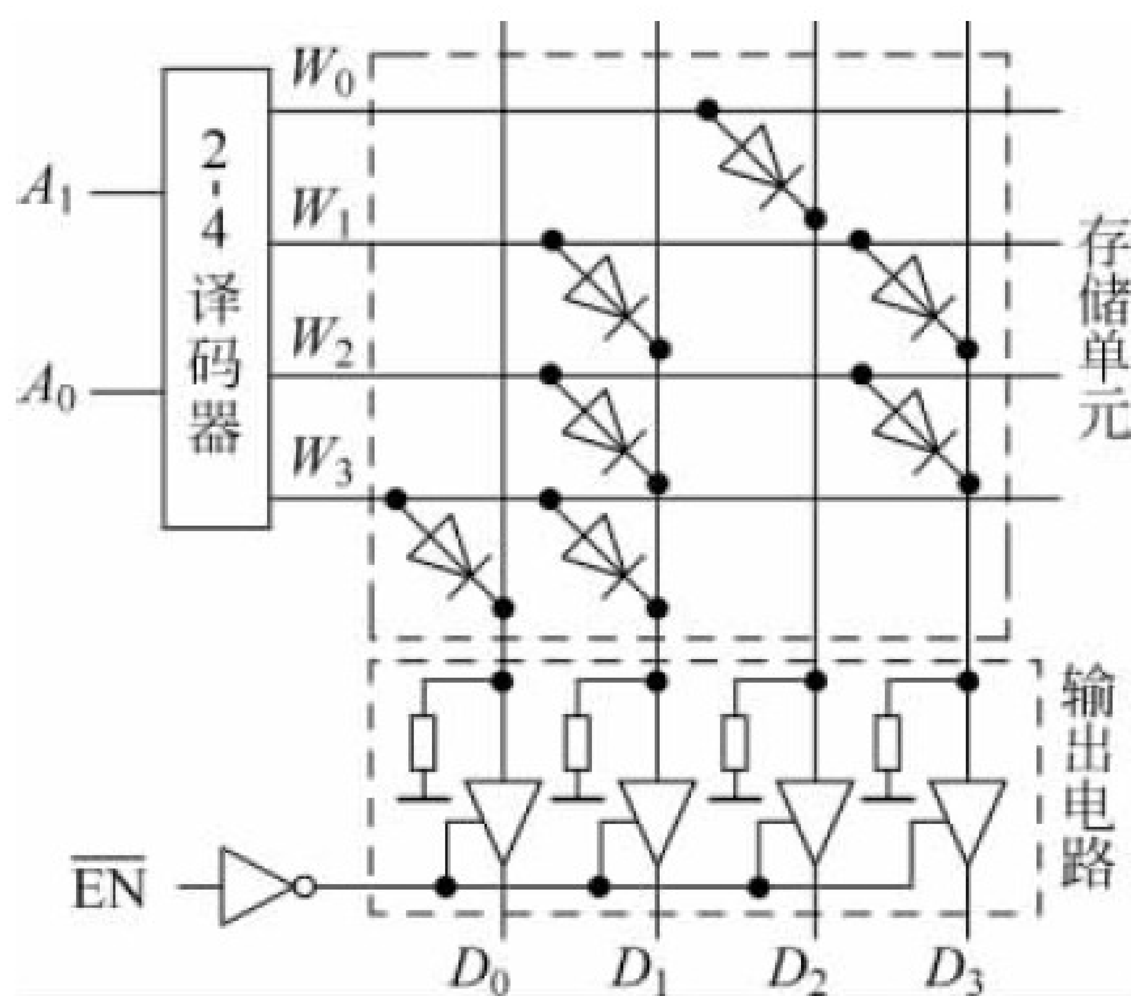


图 7.5.1 例 7.5.1 的图

下面通过几个例题介绍用大规模器件设计组合逻辑电路的方法。

【例 7.5.2】 用 2716 实现下面的函数(x_2、x_1的取值范围均为 0～3 的正整数):

$$y_2 = x_2^2 \quad y_1 = x_1^2 + 1$$

解 (1) 分析题意,做出真值表。x_2、x_1的取值范围均为 0～3 的正整数,故可用 2 位二进制数来表示;可算出 y_2的最大值为 9,y_1的最大值为 10,均可用 4 位二进制数来表示。

令 x_{21}、x_{20}表示 x_2;x_{11}、x_{10}表示 x_1;y_{23}、y_{22}、y_{21}、y_{20}表示 y_2;y_{13}、y_{12}、y_{11}、y_{10}表示 y_1;则 y_2、y_1的真值表如表 7.5.4 和表 7.5.5 所示。

表 7.5.4 真值表 1

x_{21}	x_{20}	y_{23}	y_{22}	y_{21}	y_{20}
0	0	0	0	0	0
0	1	0	0	0	1
1	0	0	1	0	0
1	1	1	0	0	1

表 7.5.5 真值表 2

x_{11}	x_{10}	y_{13}	y_{12}	y_{11}	y_{10}
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	1
1	1	1	0	1	0

(2) 设置输入输出。2716 有 11 个地址输入端,而本题只有 4 个输入变量,可令 A_{10}～A_4为 0,A_3～A_0作为输入变量,令 $x_{21}=A_3$,$x_{20}=A_2$,$x_{11}=A_1$,$x_{10}=A_0$。

2716 有 8 个数据输出端 D_7～D_0,本题正好有 8 个输出变量,可令 $y_{23}=D_7$,$y_{22}=D_6$,$y_{21}=D_5$,$y_{20}=D_4$,$y_{13}=D_3$,$y_{12}=D_2$,$y_{11}=D_1$,$y_{10}=D_0$。

则表 7.5.4 和表 7.5.5 可用如表 7.5.6 和表 7.5.7 所示的真值表来表示。

表 7.5.6　真值表 3

A_3　A_2	D_7　D_6　D_5　D_4
0　0	0　0　0　0
0　1	0　0　0　1
1　0	0　1　0　0
1　1	1　0　0　1

表 7.5.7　真值表 4

A_1　A_0	D_3　D_2　D_1　D_0
0　0	0　0　0　1
0　1	0　0　1　0
1　0	0　1　0　1
1　1	1　0　1　0

(3) 编写程序。由表 7.5.6 和表 7.5.7,2716 的 0～3 号存储单元的内容用十进制表示分别为 1、2、5、10(当地址为 0～3 时,$A_3A_2=00$,故存储单元高 4 位内容用十进制表示为“0”;低 4 位存储单元内容为表 7.5.7 中规定内容)。

4～7 号存储单元的内容用十进制表示分别为 17、18、21、26(当地址为 4～7 时,$A_3A_2=01$,故存储单元高 4 位内容用十进制表示为“1”;低 4 位存储单元内容为表 7.5.7 中规定内容)。

8～11 号存储单元的内容用十进制表示分别为 65、66、69、74(当地址为 8～11 时,$A_3A_2=10$,故存储单元高 4 位内容用十进制表示为“4”;低 4 位存储单元内容为表 7.5.7 中规定内容)。

12～15 号存储单元的内容用十进制表示分别为 145、146、149、154(当地址为 12～15 时,$A_3A_2=11$,故存储单元高 4 位内容用十进制表示为“9”;低 4 位存储单元内容为表 7.5.7 中规定内容)。

(4) 最终实现。将(3)中的 16 个数据写入 2716 的前 16 个单元,按图 7.5.2 连接电路即可。

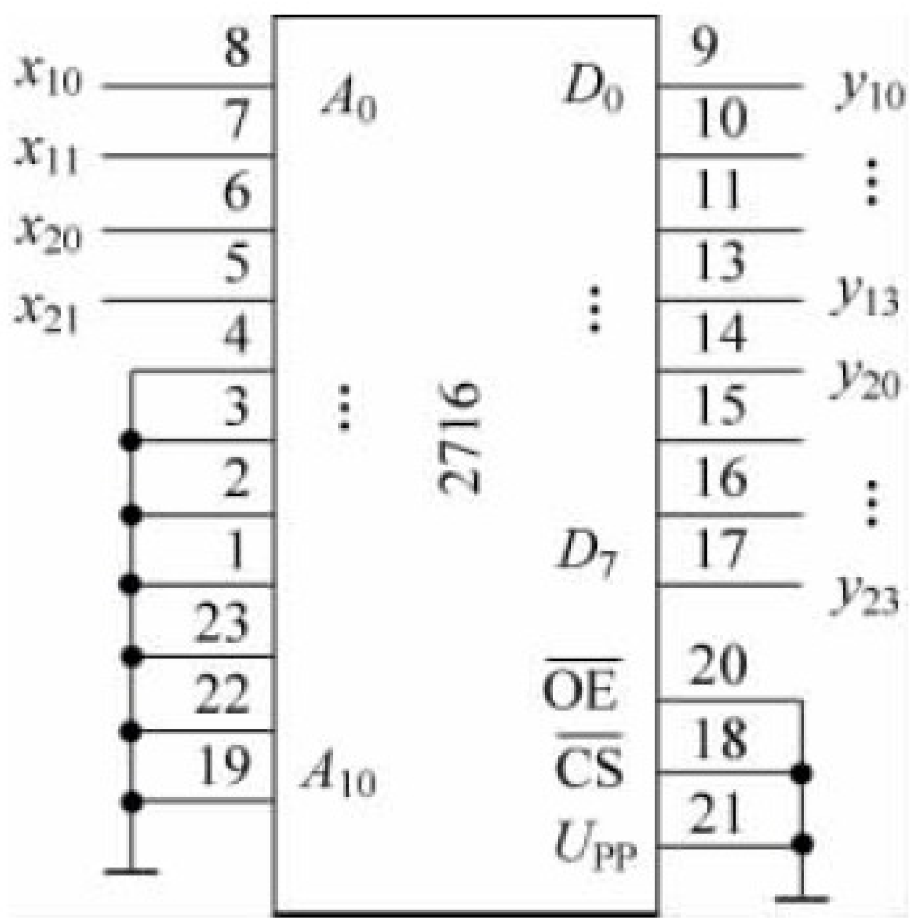


图 7.5.2　例 7.5.2 的图

【例 7.5.3】　用 2716 实现下面的函数。

$$Y_1 = A\bar{B}C + \bar{A}\bar{C}D + A\bar{C},\quad Y_2(A,B,C,D) = \sum_m (0,2,3,4,7,9)$$

解　(1) 分析题意,做出真值表。由函数表达式可作出如表 7.5.8 所示的真值表。

(2) 设置输入输出。2716 有 11 个地址输入端,而本题只有 4 个输入变量,可令 A_{10}～A_4 为 0,A_3～A_0 作为输入变量,令 $A=A_3$,$B=A_2$,$C=A_1$,$D=A_0$。

2716 有 8 个数据输出端 D_7～D_0,本题只需要 2 个输出变量,可令 $Y_2=D_1$,$Y_1=D_0$。

(3) 编写程序。由表 7.5.8,2716 的前 16 个存储单元的内容用十进制表示分别为 2、1、2、2、2、1、0、2、1、3、1、1、1、1、0、0。

表 7.5.8　例 7.5.3 的真值表 1

A	B	C	D	Y_2	Y_1
0	0	0	0	1	0
0	0	0	1	0	1
0	0	1	0	1	0

续表

A	B	C	D	Y_2	Y_1
0	0	1	1	1	0
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	0	1	1	1
1	0	1	0	0	1
1	0	1	1	0	1
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	0
1	1	1	1	0	0

(4) 最终实现。将(3)中的16个数据写入2716的前16个单元，按图7.5.3连接电路即可。

【例7.5.4】 如图7.5.4所示电路，2716的前16个存储单元的内容用十进制表示分别为2、0、2、1、0、2、3、0、0、1、2、0、3、2、2、3，分析该电路实现的逻辑功能。

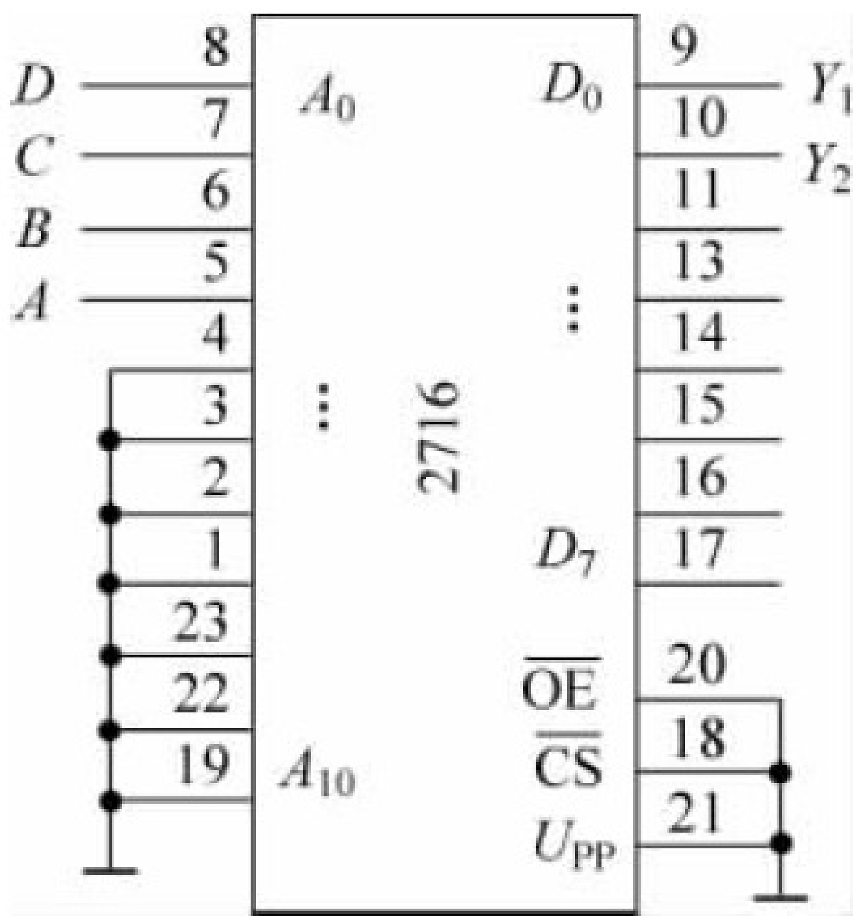


图7.5.3 例7.5.3的图

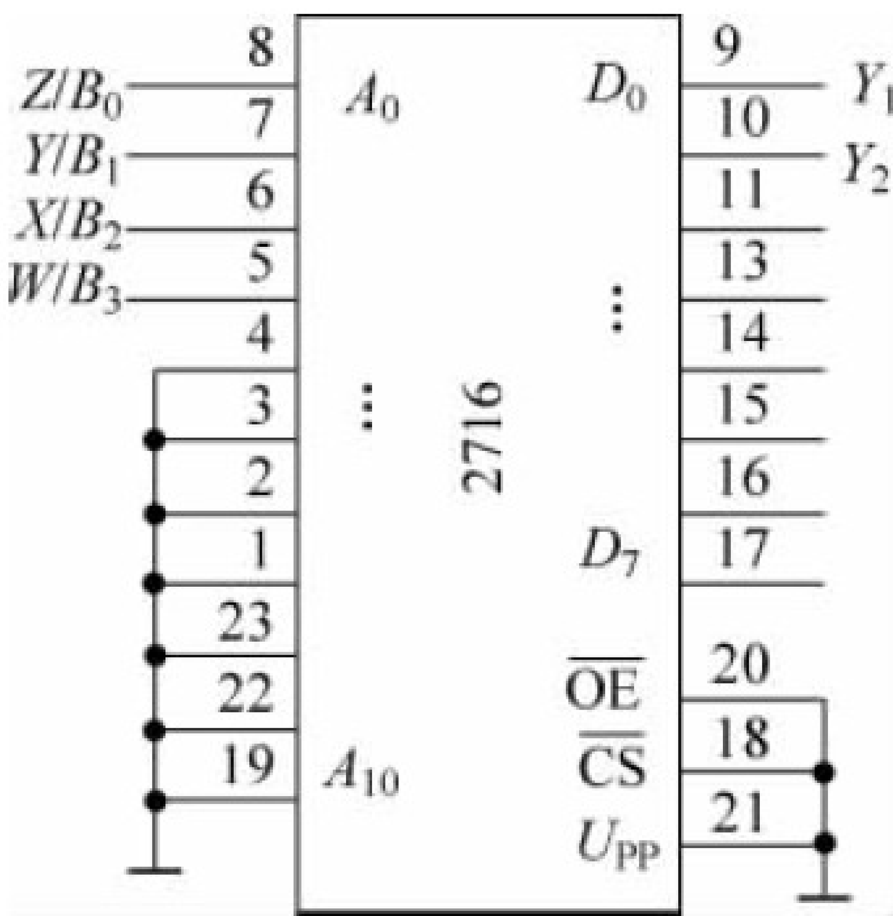


图7.5.4 例7.5.4的图

解 (1) 由ROM单元内容写出其真值表。由前16个单元内容可作出如表7.5.9所示的真值表。

表7.5.9 例7.5.4的真值表

A_3	A_2	A_1	A_0	D_1	D_0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	0	0

续表

A_3	A_2	A_1	A_0	D_1	D_0
0	1	0	1	1	0
0	1	1	0	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	1	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	1	1

(2) 分析真值表。先看 D_0，将 $A_3A_2A_1A_0$ 视为4位非零二进制数，则输出 $D_0=1$ 表示4位非零二进制数 $B_3B_2B_1B_0$ 能被3整除。

再看 D_1，令 $W=A_3$，$X=A_2$，$Y=A_1$，$Z=A_0$，则输出 $D_1=1$ 表示输血者的血型与受血者血型吻合，可以输血。血型编码如表7.5.10所示。

表 7.5.10　编码表

输　血	受　血
A 型 $\overline{W}\overline{X}$	A 型 $\overline{Y}\overline{Z}$
B 型 $\overline{W}X$	B 型 $\overline{Y}Z$
AB 型 $W\overline{X}$	AB 型 $Y\overline{Z}$
O 型 WX	O 型 YZ

通过上面几个例题不难看出，采用大规模逻辑器件设计组合逻辑电路十分方便，且可大大节省元件，可靠性更高，所以应用十分广泛。

复习与练习

7.5.1　RAM 属于时序电路还是属于组合电路？

7.5.2　在如图7.5.2所示电路中，若 $\overline{OE}$ 接1，则该ROM电路是否还具有组合逻辑电路性质？

习题

7-1　填空题

1. 通常把存储器的每个________称为一个“字”。存储器所具有的地址线的________反映了存储器的字数。存储器的字数反映了存储器的存储单元的多少。显然，具有 m 根地址线的存储器的字数为________。

2. 存储器是用于存储________的存储设备，从集成工艺的角度，存储器可分为________存储器和________存储器两种。从读写方式的角度，存储器可分为________、________两种。其中的________需要刷新，以避免存储信号的丢失。

3. 叠栅注入 MOS 管简称 SIMOS，为________采用的存储器件。如果浮置栅上没有________，则存储单元数据为0。当浮置栅充有________时，存储单元数据为1。

7-2 分析应用题

1. 有一个存储容量为1024的存储矩阵，排列成16行和64列，它需要一个什么样的行、列地址译码器？可否采用一个二进制译码器进行全地址译码？

2. 容量为256×1的RAM有多少根地址输入线？有多少根字线和位线？若要用256×1的RAM扩展成1024×4的RAM，需要多少个256×1的RAM芯片？

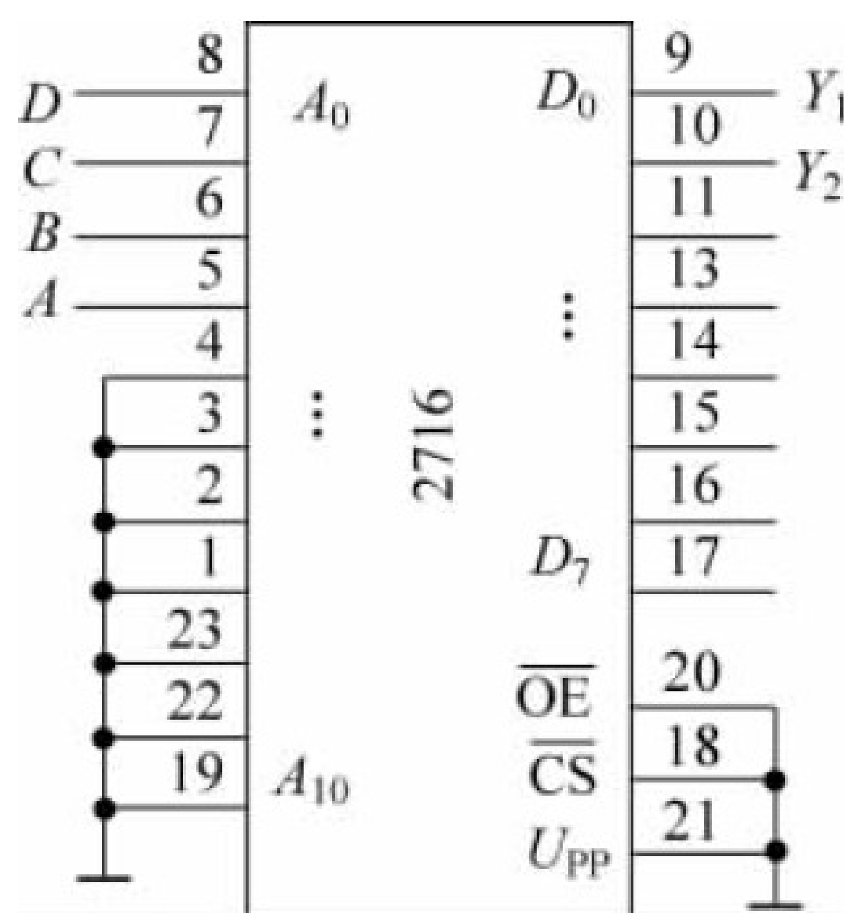
3. 用2114实现一个4K×4的存储系统。

4. 用2114实现一个1K×16的存储系统。

5. 用2114实现一个2K×8的存储系统。

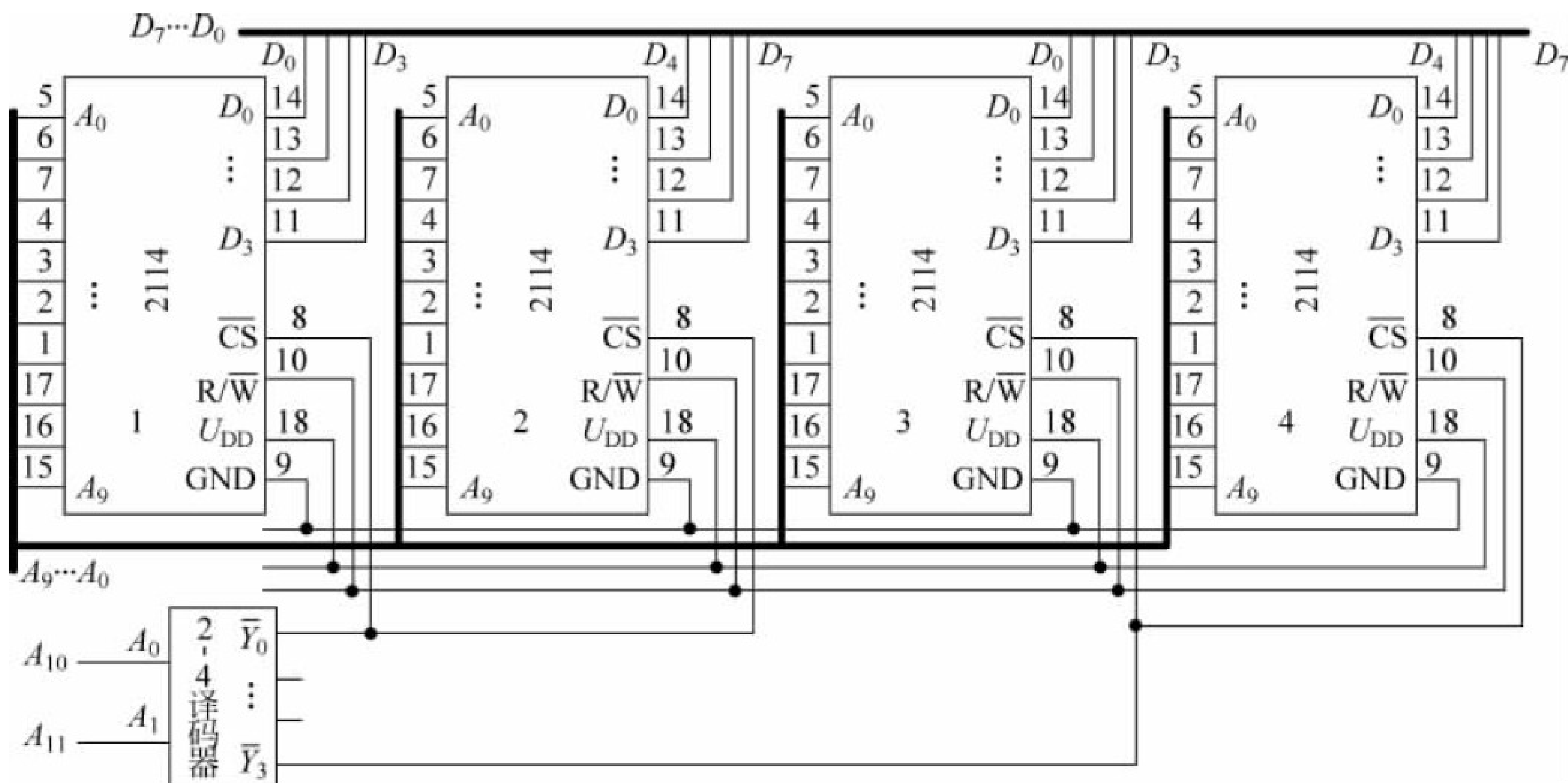
6. 用2716结合译码器实现一个8K×16的存储系统。

7. 在如题图7.1所示电路中，2716的前16个存储单元的内容用十进制表示分别为2、1、3、1、0、2、3、3、3、1、2、3、3、2、2、3，分析该电路实现的逻辑功能。



题图 7.1

8. 分析如题图7.2所示存储系统的地址空间。



题图 7.2

9. 写出如题图7.3所示ROM所表示的逻辑函数(地址译码器的有效输出电平为高电平)。

10. 试用2716实现两个1位二进制数的全加运算。

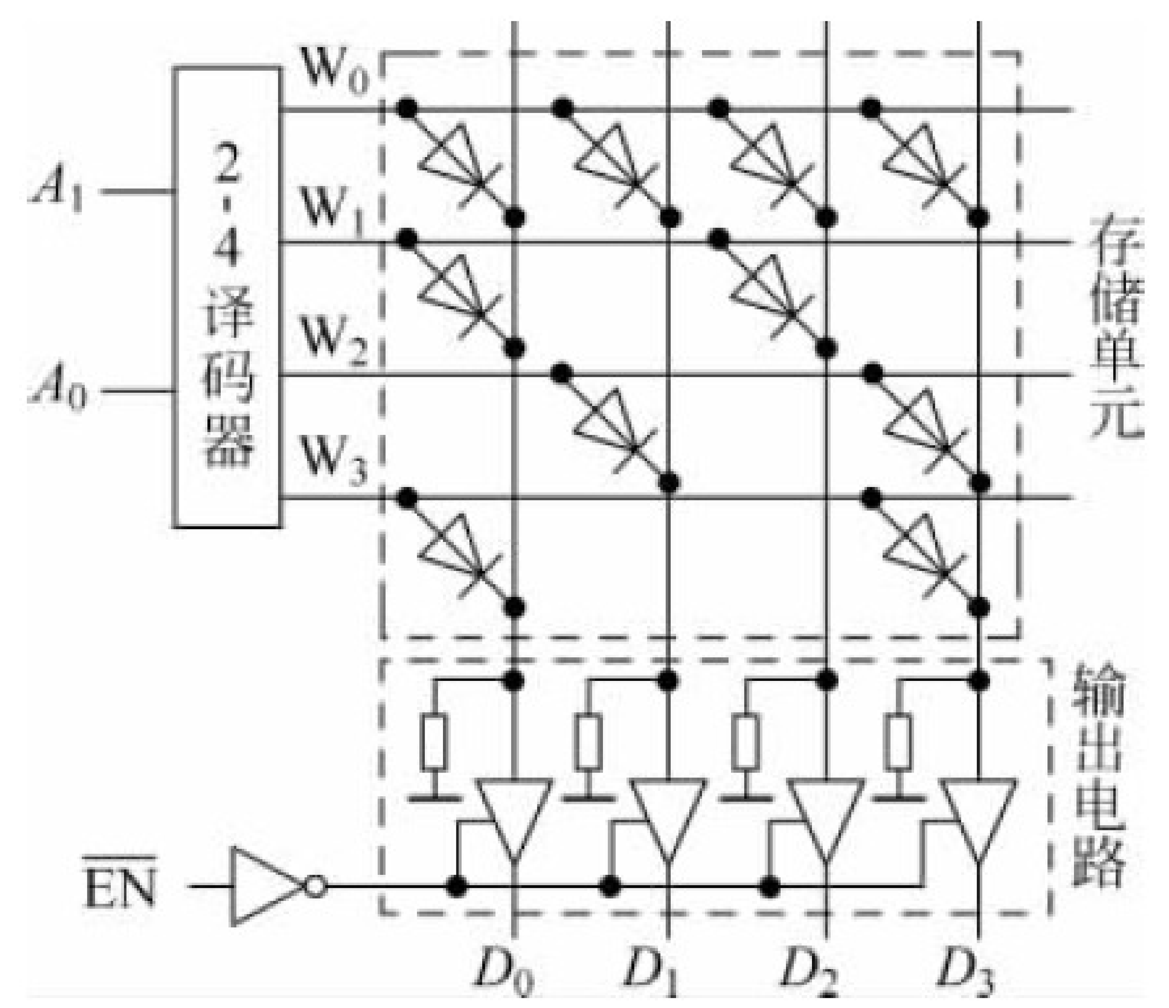
11. 用2716实现下面的函数：

$$Y_1 = A\overline{B} + \overline{A}\overline{C}D + AB\overline{C}$$

$$Y_2(A,B,C,D) = \sum_m (0,3,4,6,9,10)$$

12. 用74LS161结合2716实现一个4位顺序正脉冲输出的脉冲发生器。

13. 用74LS161结合2716实现一个"1110"序列信号发生器。

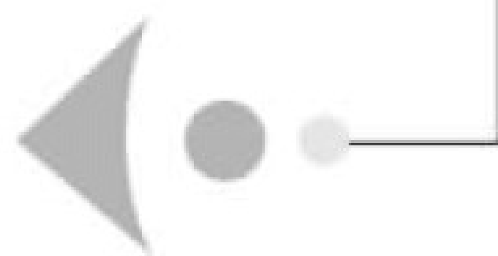


题图　7.3



第8章

模/数转换器与数/模转换器



本章要点：

模/数转换器与数/模转换器是利用数字系统处理模拟信号的基本器件。学习本章应理解数/模转换、模/数转换的基本原理及转换过程。在此基础上，理解权电阻网络、倒T形电阻网络等典型数/模转换电路的特点，以及并联比较型、反馈比较型等典型模/数转换电路的特点。



8.1 概述

常见的非电物理量，如压力、流量、液位、温度、光通量等，都可以通过相应的换能器或传感器、敏感器等变换为随时间连续变化的电信号，称为模拟量。如果要将这些模拟量送到数字系统中进行处理，必须先将这些模拟量转换成数字量，实现这一功能的电路称为模/数转换器，简称 ADC(Analog Digital Converter)或 A/D 转换器。

当然，模/数转换器只是简单地将一个输入电压信号转换为一个数字信号。这个数字信号本身不具有实际意义，仅仅表示一个相对大小，因此，模/数转换器需要一个参考量作为转换的标准，比较常见的参考标准为最大的可转换信号大小，输出的数字量则表示输入信号相对于参考信号的大小。

在生产实际中，往往还需要将经过数字系统分析处理后的结果(数字量)变换为相应的模拟量，实现这一功能的电路称为数/模转换器，简称 DAC(Digital Analog Converter)或 D/A 转换器。

A/D 转换器和 D/A 转换器广泛应用于数据传输系统、自动测试设备、医疗信息处理、电视信号的数字化、图像信号的处理和识别、数字通信和语音信息处理等数字系统中，是构成模拟信号数字处理系统的必需器件。

D/A 转换器和 A/D 转换器类型很多。常用的 D/A 转换器有权电阻网络 D/A 转换器、

权电流型 D/A 转换器、权电容网络 D/A 转换器、倒 T 形电阻网络 D/A 转换器等。常用的 A/D 转换器包括直接型 A/D 转换器和间接型 A/D 转换器两大类。直接型 A/D 转换器将获取的模拟信号直接转换为数字信号，具有更快的转换速度。间接型 A/D 转换器先将获取的模拟信号转换为频率等中间量，通过中间量获得最终的数字量。显然，间接型 A/D 转换器转换速度要比直接型慢很多，先将原始信号转换为中间量的目的是为了获得更更高的转换精度。

8.2 数/模转换器



D/A 转换器类型很多，本节主要介绍 D/A 转换的一般过程及典型电路的基本特点。

8.2.1 数/模转换器的基本原理

1. 转换特性

可以将 D/A 转换器看成一个译码器，它是将输入的二进制数字信号 D（或称为编码信号）转换（翻译）成模拟信号，并以电压或电流的形式输出。图 8.2.1 所示为 D/A 转换器输入、输出关系框图，$D_0 \sim D_{n-1}$ 为输入的 n 位二进制数，U_O 或 I_O 为与输入二进制数成比例的输出电压或电流。

如图 8.2.2 所示为 3 位二进制输入时的 D/A 转换特性。它表示 3 位二进制代码的数字信号经过 D/A 转换器后的输出模拟（电压）信号的对应关系。从图中可见，每一个二进制代码的编码数字信号通过位权的运算，都可翻译成一个相对应的十进制数值。必须指出，相邻两个编码信号转换出来的数值是不连续的值，它们之间的差值由最低码位所代表的位权确定。这是信息所能分解的最小值。对于 3 位二进制代码，该差值一般为 1/8×满值。

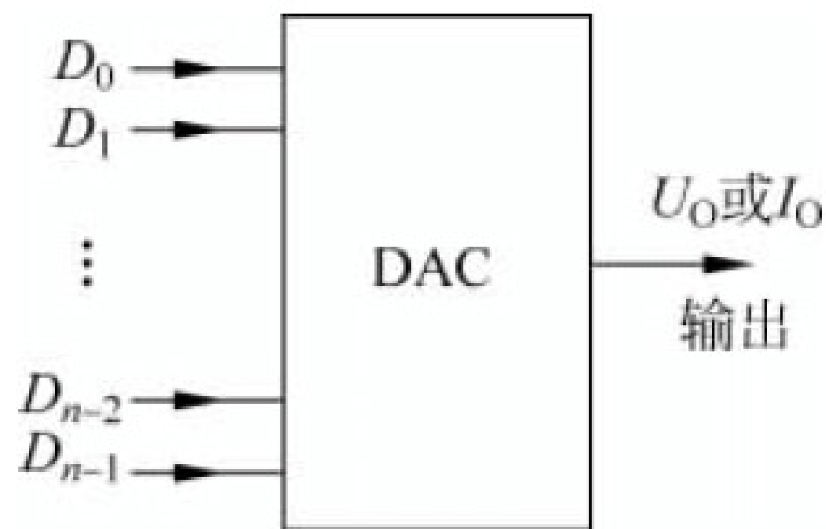


图 8.2.1 输入、输出框图

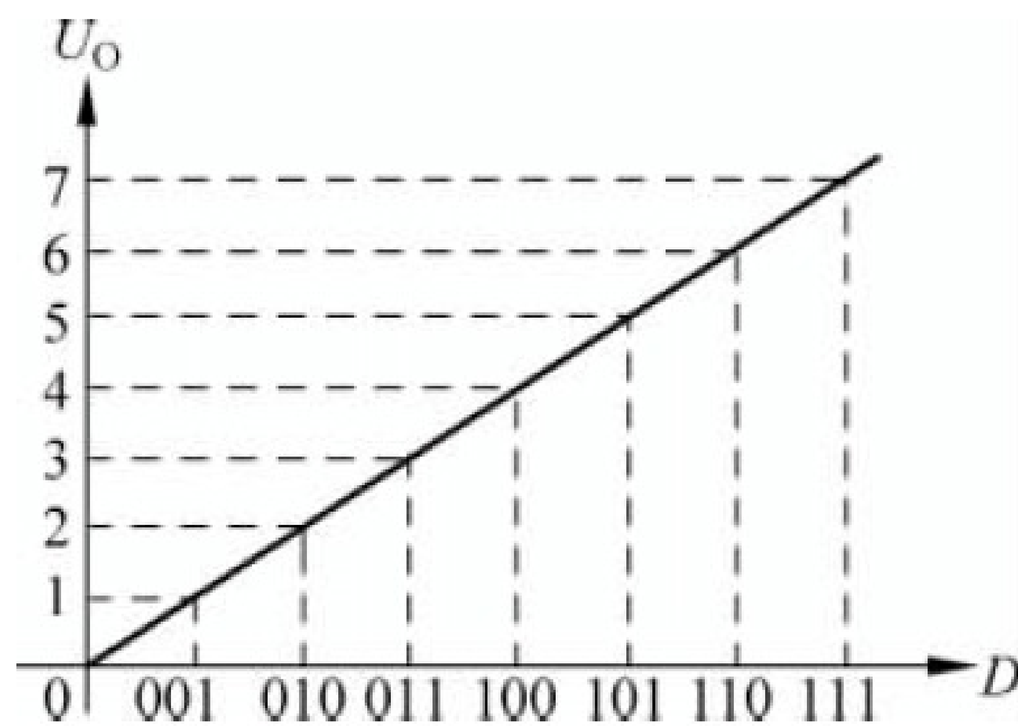


图 8.2.2 3 位二进制 D/A

显然，如果增加数字编码信号的位数，可以使相邻两个编码信号转换输出的差值减小。例如，输入的是 10 位二进制代码，其输出电压可能的最小变化一般为满值输出的 1/1024。这说明输入信号的位数越多，输出的模拟信号越接近连续的模拟信号，从而转换精度就越高。

转换精度常用分辨率和转换误差来描述。转换器所能分辨出的最小输出电压与最大

输出电压之比称为分辨率，n 位 D/A 转换器的分辨率为 $1/(2^n-1)$。分辨率为 D/A 转换器的理论精度，其实际转换误差还与其他实际因素有关。

2. 转换原理

如图 8.2.3 所示为 D/A 转换器的转换原理方框图。它是先将需要转换的数字信号以并行输入（或者串行输入）的方式存储在数字寄存器中，然后由寄存器并行输出的每一位数字信号驱动一个数字位模拟开关，又通过模拟开关将参考电压按位权关系加到解码网络，这时输出的模拟电压刚好与该位数码所代表的的数值相对应。

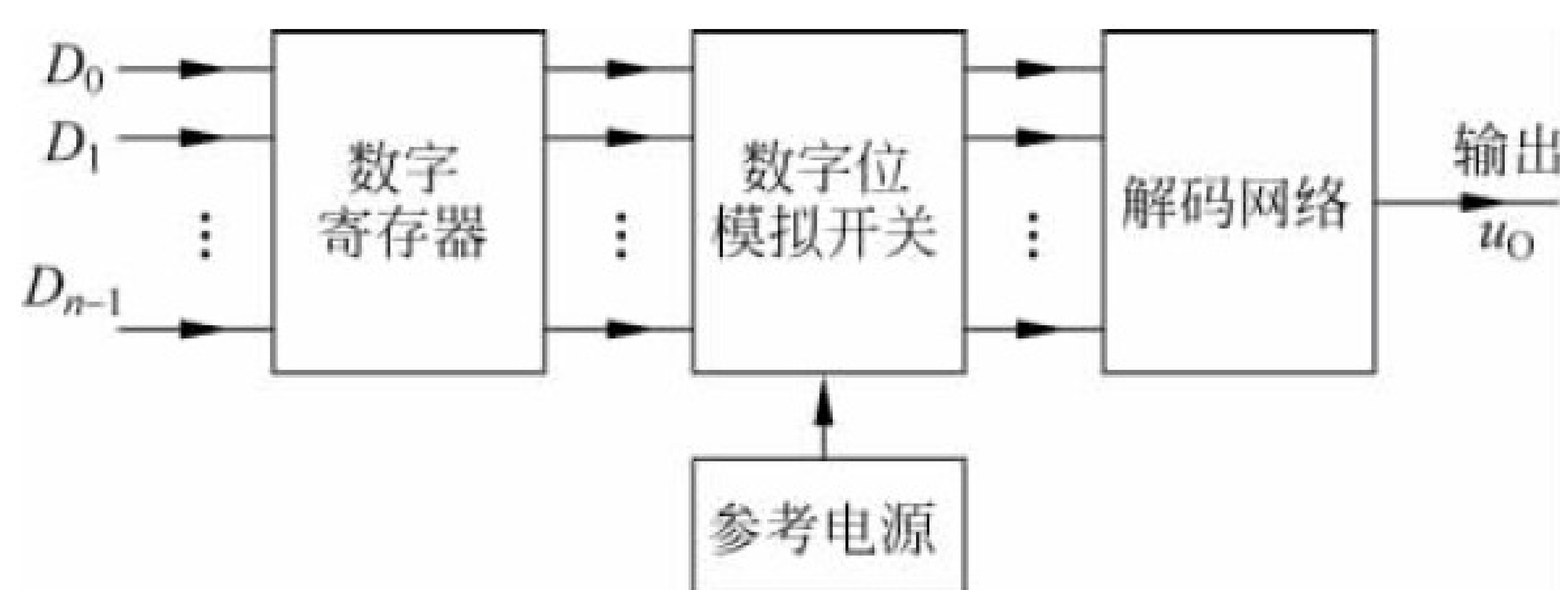


图 8.2.3　D/A 转换原理方框图

8.2.2　权电阻网络数/模转换器

1. 电路组成

如图 8.2.4 所示为权电阻网络 D/A 转换器。它由以下部分组成。

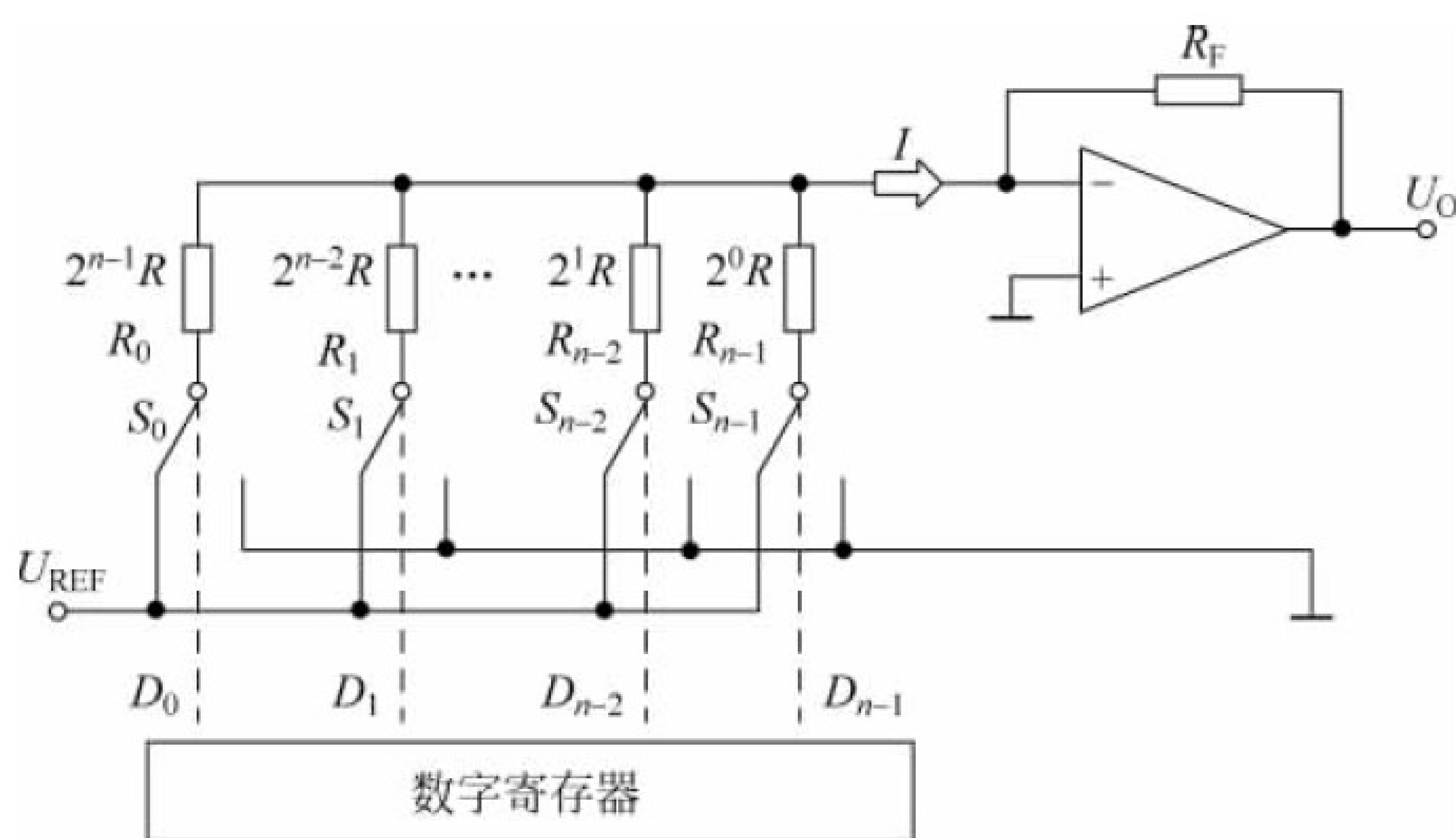


图 8.2.4　权电阻 D/A 转换器

（1）数字寄存器。在数字寄存器中存放着所需要转换的二进制代码的数字信号。

（2）模拟电子开关 S_i。模拟电子开关 S_{n-1}、S_{n-2}、…、S_1、S_0 分别受数字寄存器输出的数码控制。若 $D_i=1$，则 S_i 将电阻网络中相应位的电阻 R_i 和参考电压 U_{REF} 接通；若 $D_i=0$，则 S_i 将相应位的电阻 R_i 和地接通。

（3）权电阻网络。电阻网络中，每个电阻的阻值与相应位的位权成反比。即对于 n 位

二进制代码,权电阻网络由 n 个电阻组成。电阻网络中各支路的电阻阻值 $R_0 \sim R_{n-1}$,按二进制数的位权大小成比例地减小。例如,输入二进制代码的最高位 D_{n-1},其位权值为 2^{n-1},其权电阻阻值为 $R_{n-1}=2^{n-1-(n-1)}R=2^0 R$;最低位 D_0,其位权值为 2^0,其权电阻阻值为 $R_0=2^{n-1-0}R=2^{n-1}R$。故可写出第 i 位的权电阻阻值为 $R_i=2^{n-1-i}R$。

(4) 运算放大器和参考电压 U_{REF}。图 8.2.4 中,运算放大器接成带负反馈的求和放大器。电压 U_{REF} 称为参考电压,要求精确、稳定。

2. 工作原理

当输入数码 $D_i=0$ 时,电子开关 S_i 将电阻 $R_i=2^{n-1-i}R$ 与地接通,则电流 $I_i=0$。

当输入数码 $D_i=1$ 时,电子开关 S_i 将电阻 R_i 与参考电源 U_{REF} 接通,则电流 I_i 为

$$I_i=\frac{U_{\text{REF}}}{R_i}=\frac{U_{\text{REF}}}{2^{n-1-i}R}=\frac{U_{\text{REF}}}{2^{n-1}R}2^i \tag{8.2.1}$$

故 I_i 可以写成一般式,即

$$I_i=\frac{U_{\text{REF}}}{2^{n-1}R}\times 2^i \times D_i \tag{8.2.2}$$

根据式(8.2.2)可以写出总电流 i 为

$$I=\sum_{i=0}^{n-1} I_i=\sum_{i=0}^{n-1}\frac{U_{\text{REF}}}{2^{n-1}R}\times 2^i\times D_i=\frac{U_{\text{REF}}}{2^{n-1}R}\sum_{i=0}^{n-1}D_i\times 2^i \tag{8.2.3}$$

由运算放大器的性质可求出输出电压(D/A 转换完成后的模拟输出电压)U_{O} 为

$$U_{\text{O}}=-IR_{\text{F}}=-\frac{U_{\text{REF}}}{2^n}\frac{2R_{\text{F}}}{R}\sum_{i=0}^{n-1}D_i\times 2^i \tag{8.2.4}$$

若取转换比例系数 $\frac{2R_{\text{F}}}{R}=1$,则式(8.2.4)可写成

$$U_{\text{O}}=-\frac{U_{\text{REF}}}{2^n}(D_{n-1}\times 2^{n-1}+\cdots+D_1\times 2^1+D_0\times 2^0) \tag{8.2.5}$$

【例 8.2.1】 已知一个 4 位权电阻 D/A 输入的 4 位二进制数码为 $D_3D_2D_1D_0=1011$,参考电源 $U_{\text{REF}}=-8\text{V}$,转换比例系数为 1。求转换后的模拟信号由电压 U_{O}。

解 根据式(8.2.5)可求出

$$U_{\text{O}}=-\frac{-8}{2^4}(1\times 2^3+0\times 2^2+1\times 2^1+1\times 2^0)=5.5\text{V}$$

图 8.2.4 所示权电阻 D/A 的优点是电路结构比较简单,所用的电阻元件数较少;缺点是各位电阻的阻值相差较大,当位数较多时要保证很高的阻值精度就比较困难。例如,当输入信号增加到 8 位时,若取权电阻网络中最小的电阻为 $R=10\text{k}\Omega$,则最大的电阻阻值将达到 $2^7R=1.28\text{M}\Omega$,两者相差 128 倍以上,要保证很高的精度十分困难。

8.2.3 倒 T 形电阻网络数/模转换器

如图 8.2.5 所示为 4 位二进制代码倒 T 形电阻网络 D/A 转换器。4 位二进制代码

$D_3 \sim D_0$由数字寄存器输出，并控制该位的模拟电子开关。

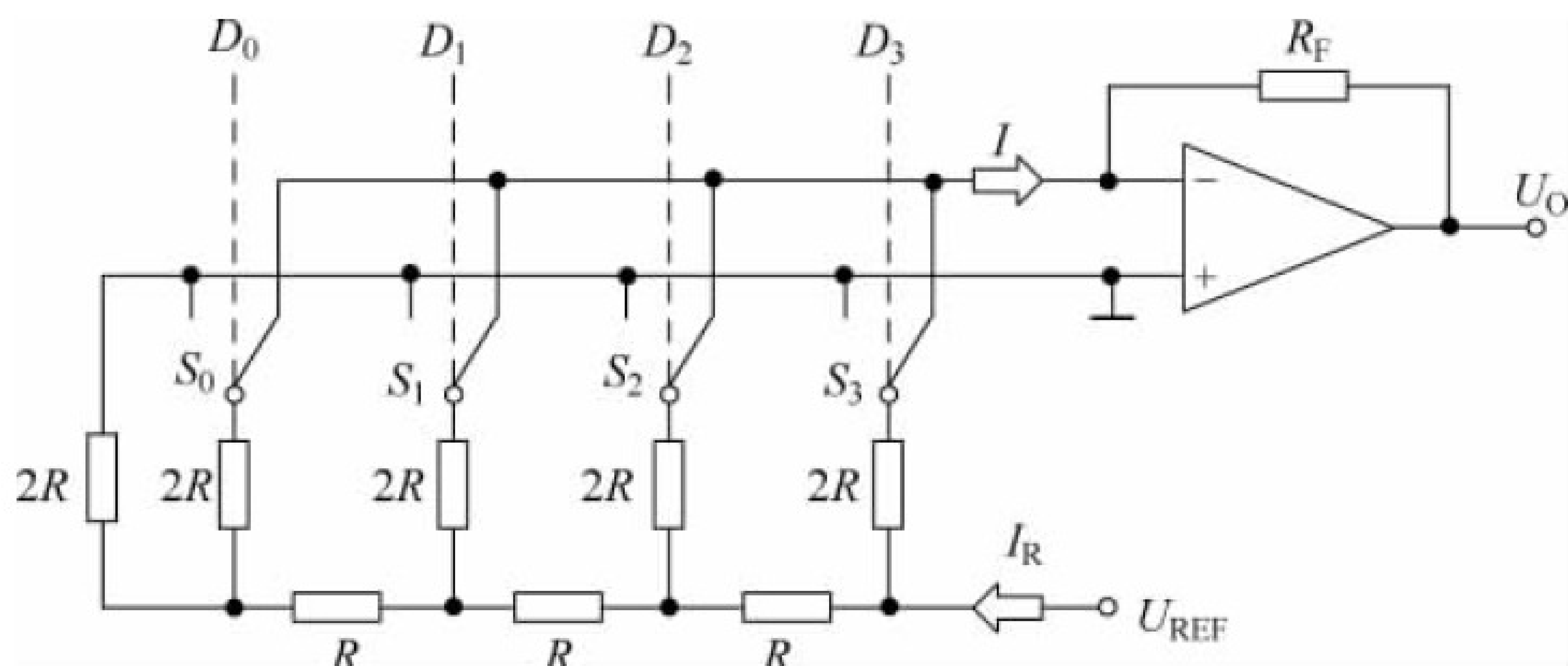


图 8.2.5　倒 T 形电阻网络 D/A 转换器

为了克服权电阻网络 D/A 转换器中权电阻网络电阻阻值相差较大的缺点，倒 T 形电阻网络中只用了 R、$2R$ 两种阻值的电阻。这不仅克服了电阻阻值相差太大的缺点，而且给集成电路的设计和制作带来了很大的方便。

在如图 8.2.6 所示的 4 位倒 T 形电阻网络中，AA′、BB′、CC′、DD′左边部分的等效电阻均为 R。所以，无论数 $D_i=0$ 还是 $D_i=1$，流过每个开关的电流始终不变。

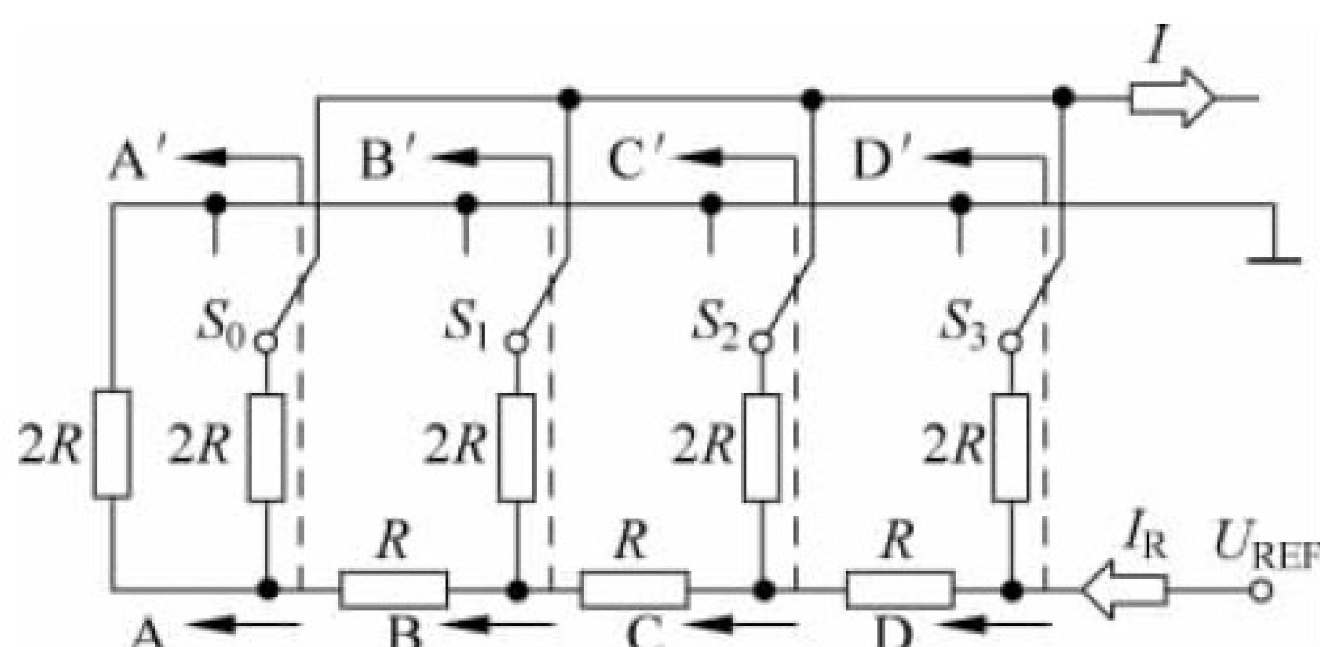


图 8.2.6　倒 T 形电阻网络

显然，电流 $I_R=U_{REF}/R$。然后利用分流公式可以求出

$$I_{S_3} = \frac{1}{2}I_R = \frac{U_{REF}}{R\times 2^1}$$

$$I_{S_2} = \frac{1}{2}I_{S_3} = \frac{U_{REF}}{R\times 2^2}$$

$$I_{S_1} = \frac{1}{2}I_{S_2} = \frac{U_{REF}}{R\times 2^3}$$

$$I_{S_0} = \frac{1}{2}I_{S_1} = \frac{U_{REF}}{R\times 2^4}$$

又由于当 $D_i=1$ 时，S_i接到运算放大器的反相输入端；$D_i=0$ 时，S_i接地。

所以，可以写出总电流 I 为

$$\begin{aligned} I &= \frac{U_{REF}}{R2^1}D_3 + \frac{U_{REF}}{R2^2}D_2 + \frac{U_{REF}}{R2^3}D_1 + \frac{U_{REF}}{R2^4}D_0 \\ &= \frac{U_{REF}}{R2^4}(D_3\times 2^3 + D_2\times 2^2 + D_1\times 2^1 + D_0\times 2^0) \end{aligned}$$

从图 8.2.5 中可求出输出电压 U_O 为

$$\begin{aligned} U_O &= -IR_F \\ &= -\frac{R_F}{R}\frac{U_{REF}}{2^4}(D_3 \times 2^3 + D_2 \times 2^2 + D_1 \times 2^1 + D_0 \times 2^0) \end{aligned} \tag{8.2.6}$$

取转换比例系数 $\frac{R_F}{R}=1$，由式(8.2.6)可以写出 n 位倒 T 形 D/A 转换器输出模拟信号电压 U_O 的表达式为

$$U_O = -\frac{U_{REF}}{2^n}\sum_{i=0}^{n-1} D_i \times 2^i \tag{8.2.7}$$

8.2.4　其他常用数/模转换器

前面介绍的 D/A 转换器主要通过模拟开关将参考电压按位权关系加到电阻解码网络实现 D/A 转换的。其中，模拟开关视为理想。

实际的开关总是存在着一定的导通电阻和导通压降的，每个开关的导通电阻和导通压降一般存在一些细微的区别，这些无疑会影响转换精度。

将如图 8.2.4 和图 8.2.5 所示的电阻网络改用如图 8.2.7 所示的恒流源网络。每个恒流源电流的大小依次为前一个的 1/2，和输入二进制数对应位的“权”成正比，利用这种方式实现的 D/A 转换器便是权电流型的 D/A 转换器。

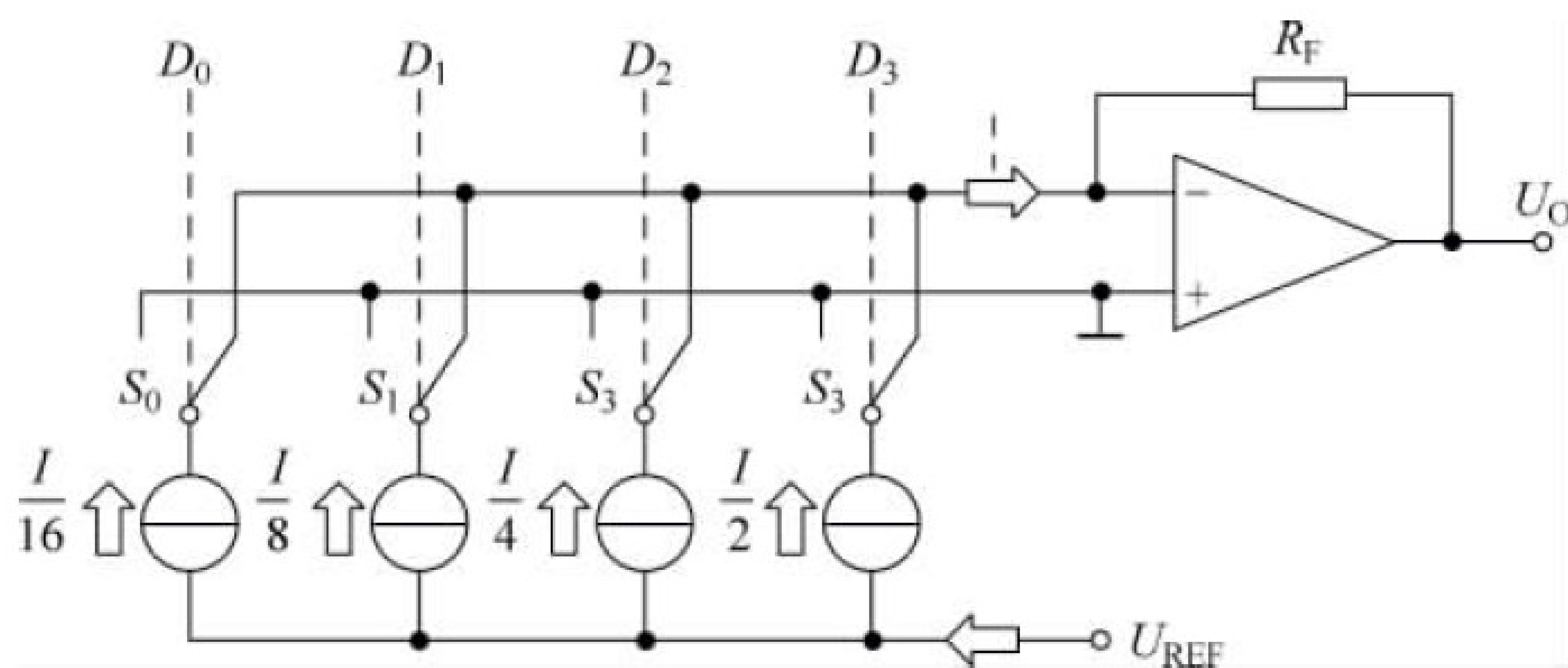


图 8.2.7　权电流型 D/A 转换器

可求出如图 8.2.7 所示 D/A 转换器的输出电压为

$$\begin{aligned} U_O &= -R_F i = -R_F\left(\frac{I}{2^1}D_3 + \frac{I}{2^2}D_2 + \frac{I}{2^3}D_1 + \frac{I}{2^4}D_0\right) \\ &= -\frac{R_F I}{2^4}(D_3 \times 2^3 + D_2 \times 2^2 + D_1 \times 2^1 + D_0 \times 2^0) \end{aligned} \tag{8.2.8}$$

可见，输出电压 U_O 正比于输入的数字量。

由于电流源具有非常高的内阻，可以认为每条支路的电流不受开关的导通和导通压降的影响，降低了 D/A 转换器对开关电路的要求，提高了转换精度。

当然，在上面的 D/A 转换器中，恒流源电路的性能对 D/A 转换器的性能有着非常关键的作用。限于篇幅，权电流型 D/A 转换器中的恒流源电路的结构及其分析方法，请读者查

阅相关书籍。

如图 8.2.8 所示为权电容网络 D/A 转换器的原理电路，是利用电容分压的原理工作的。图中，C_0、C_1、C_2、C_3的电容量按照 2 的乘方倍数递增；开关 S_0、S_1、S_2、S_3受输入的数字信号 D_0、D_1、D_2、D_3的控制。当 $D_i=1$ 时，开关 S_i接电源；$D_i=0$ 时，开关 S_i接地。

转换前，所有开关接地，全部电容器充分放电。之后，断开开关 S，将待转换数据 $D_3D_2D_1D_0$（1111）加到数据输入端，开关 S_0～S_3均接至U_{REF}一边，如图 8.2.8 所示电路的等效电路如图 8.2.9 所示。求解如图 8.2.9 所示电路稳态下的输出电压，有

$$U_O=\frac{C_0+C_1+C_2+C_3}{C_0'+C_0+C_1+C_2+C_3}U_{REF}$$
$$=\frac{(2^3+2^2+2^1+2^0)C}{(2^3+2^2+2^1+2^0+2^0)C}U_{REF}=\frac{2^4-1}{2^4}U_{REF} \tag{8.2.9}$$

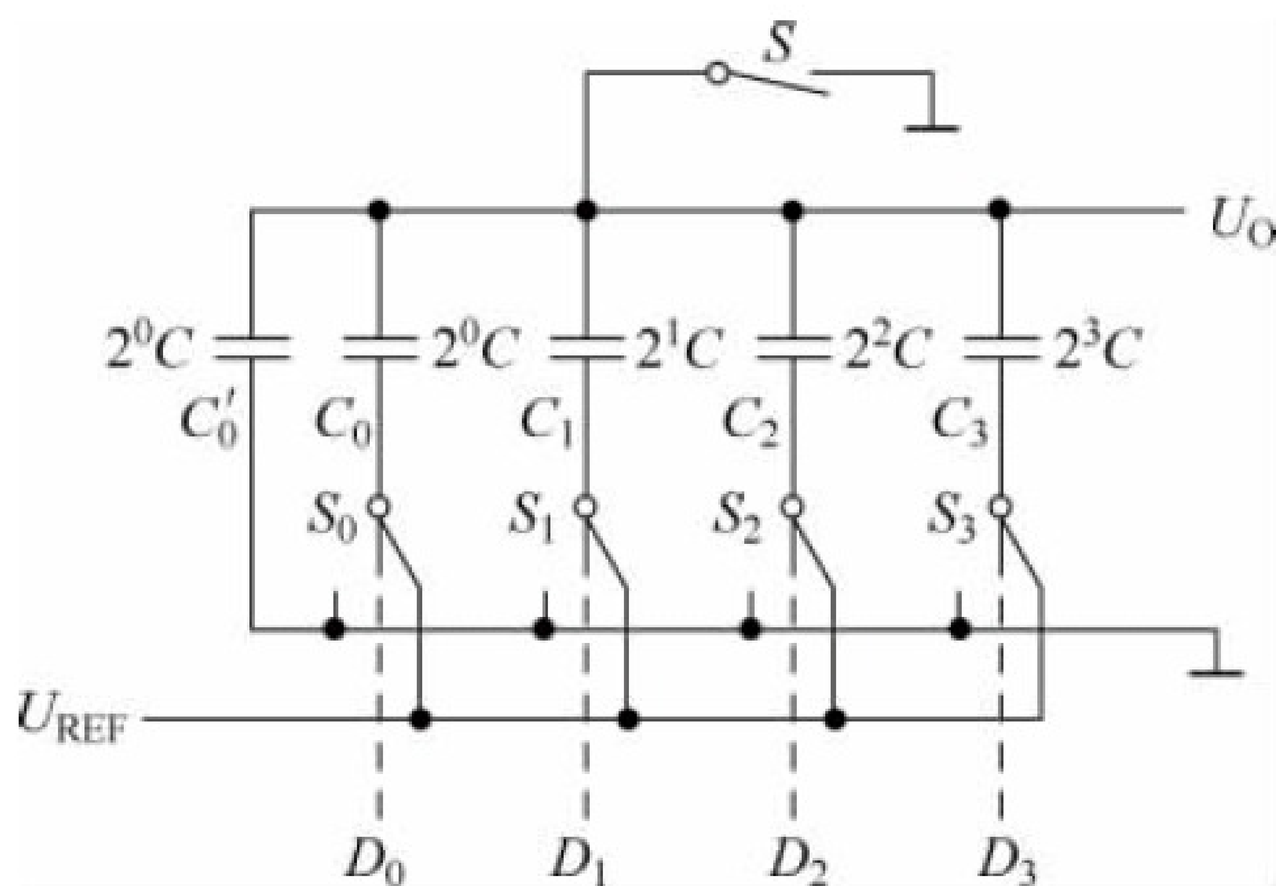


图 8.2.8　权电容网络 D/A 转换器

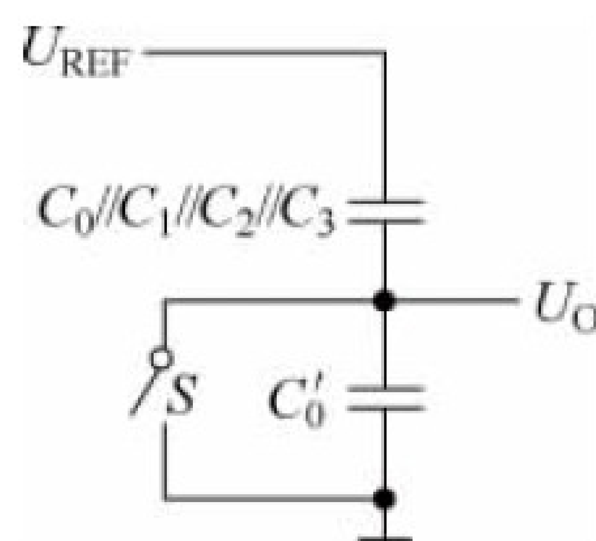


图 8.2.9　等效电路

参考上面的分析，可得出任意输入数据下输出电压

$$U_O=\frac{D_3\times2^3+D_2\times2^2+D_1\times2^1+D_0\times2^0}{2^4}U_{REF} \tag{8.2.10}$$

可见，输出电压 U_O正比于输入的数字量。

可总结权电容网络 D/A 转换器的主要特点如下：

（1）电容元件为储能器件，因此，稳态下，权电容网络不消耗功率。

（2）电容元件具有隔直流的性质，输出电压 U_O稳态值不受开关内阻、参考电压源内阻等因素的影响。

（3）输出电压的精度只与各电容器容量的比例有关。

在 MOS 集成电路中，一般直接利用器件的分布电容制作电容器，可通过精确控制电容器的尺寸实现各电容器之间电容量的严格比例关系，制作相应的集成电路十分方便。

当然，权电容网络 D/A 转换器也有明显的不足之处。当输入位数较多时，各电容器的容量相差很大，电容器的容量也相对较大，在显著增加了硅片面积的同时，显著增加了电容的充、放电时间，降低了 D/A 转换器的转换速度。

8.2.5　数/模转换器芯片实例及其典型电路

D/A 转换器是构成模拟信号数字处理系统的必需器件，相应的集成芯片非常多。倒 T 形电阻网络 D/A 转换器的主要集成电路产品有 8 位、10 位、12 位和 16 位 D/A 转换器。如图 8.2.10 所示为 10 位 D/A 转换器 AD7520 的引脚图。它采用 CMOS 型模拟开关，内部没有运算放大器；U_{DD} 为 CMOS 开关工作电源，U_{REF} 为转换器的参考电压，I_{OUT1}、I_{OUT2} 分别对应外接运算放大器的反相端及同相端。

在倒 T 形电阻网络 D/A 转换器中，各支路电流直接流入运算放大器输入端，它们之间不存在传输时间差，所以转换速度高。

AD7520 的典型接法如图 8.2.11 所示。由于 AD7520 内部反馈电阻 $R_F = R$，所以，图 8.2.11 的转换关系如下：

$$U_O = -\frac{U_{REF}}{2^{10}} \sum_{i=0}^{9} D_i \times 2^i \tag{8.2.11}$$

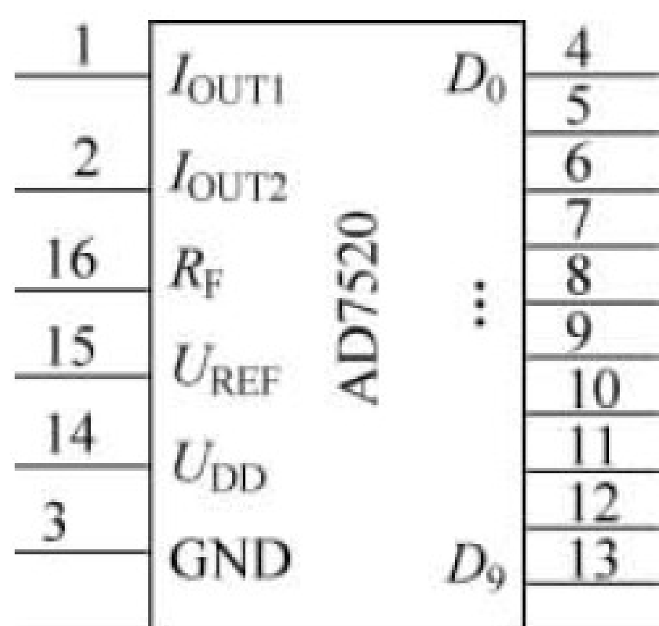


图 8.2.10　AD7520 引脚图

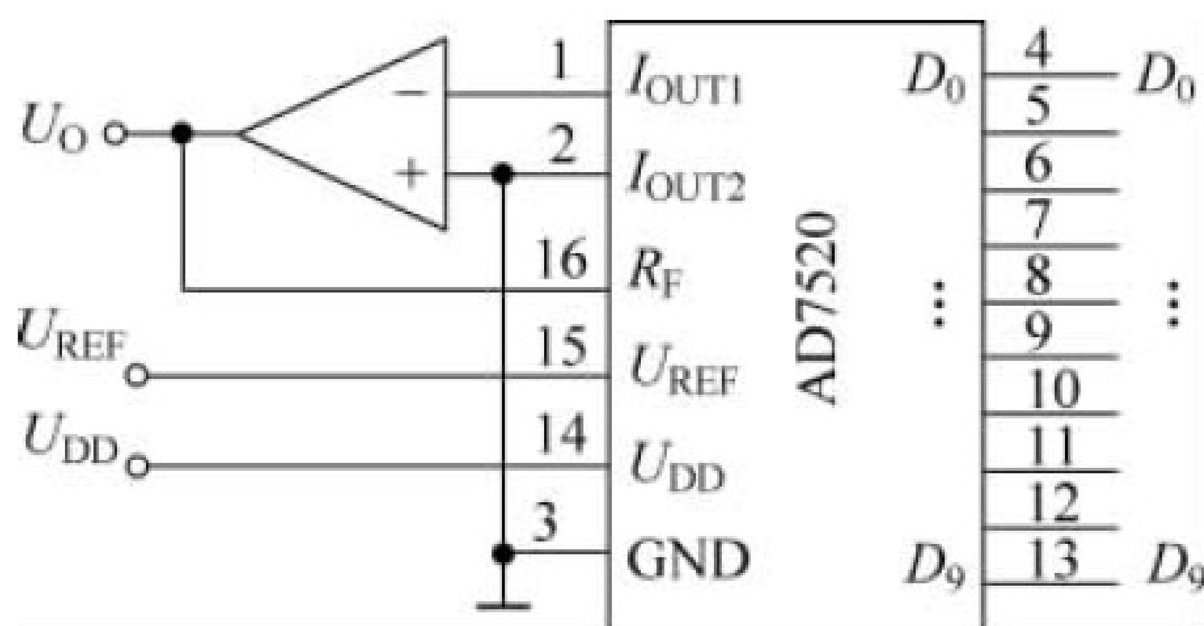


图 8.2.11　AD7520 的典型接法图

复习与思考

8.2.1　倒 T 形电阻网络 D/A 转换器与权电阻网络 D/A 转换器各有什么特点？

8.2.2　D/A 转换器能否实现模拟信号的放大功能？

8.2.3　AD7520 的理论转换精度为多少？

8.2.4　依照电路知识，电容器的充、放电路一般采用一阶 RC 电路的模型来描述，如图 8.2.8 所示权电容网络 D/A 转换器的原理电路为何没采用一阶 RC 电路的模型来描述？

8.3　模/数转换器



在 A/D 转换器中，输入的模拟信号在时间上、数值上均是连续的量，输出的数字信号在时间上、数值上均是离散的，某个时刻 A/D 转换器输出的数字信号只能是输入的连续模拟信号上的某个孤立的值，因此，A/D 转换过程要比 D/A 转换过程复杂得多。

8.3.1　模/数转换器的基本原理

进行 A/D 转换时必须在一系列选定的瞬间对输入的模拟信号采样，然后再将这些采样

值转换为输出的数字量。因此，一般 A/D 转换过程需要经过采样、保持、量化、编码 4 个步骤才能完成。有些工作过程是利用同一个电路连续进行的。例如，采样、保持就选用同一个电路完成；量化和编码也是在转换过程中同时实现的，而且所占用的时间是保持时间的一部分。

1. 采样

所谓采样，是指将一个连续变化的模拟量转换为时间上离散的模拟量。也就是说将一个在时间上连续的模拟量转换成一系列脉冲，而这些脉冲是等宽的，其幅度大小取决于采样时输入的模拟量，如图 8.3.1(b)所示。

在图 8.3.1 中，u_I为输入模拟量，CP_s为采样脉冲，u_I^* 为采样后的输出信号。所以，采样电路实际上是一个受采样脉冲控制的电子开关，如图 8.3.1(a)所示。在采样脉冲 CP_s 的脉冲宽度 t_w时间内，开关接通。此时输出 u_I^* 等于输入 u_I；而在 T_s-t_w时间内，开关断开，输出 u_I^* 为 0。于是，模拟电子开关在采样脉冲 CP_s作用下周期性地动作，u_I^* 的输出波形如图 8.3.1(b)所示。

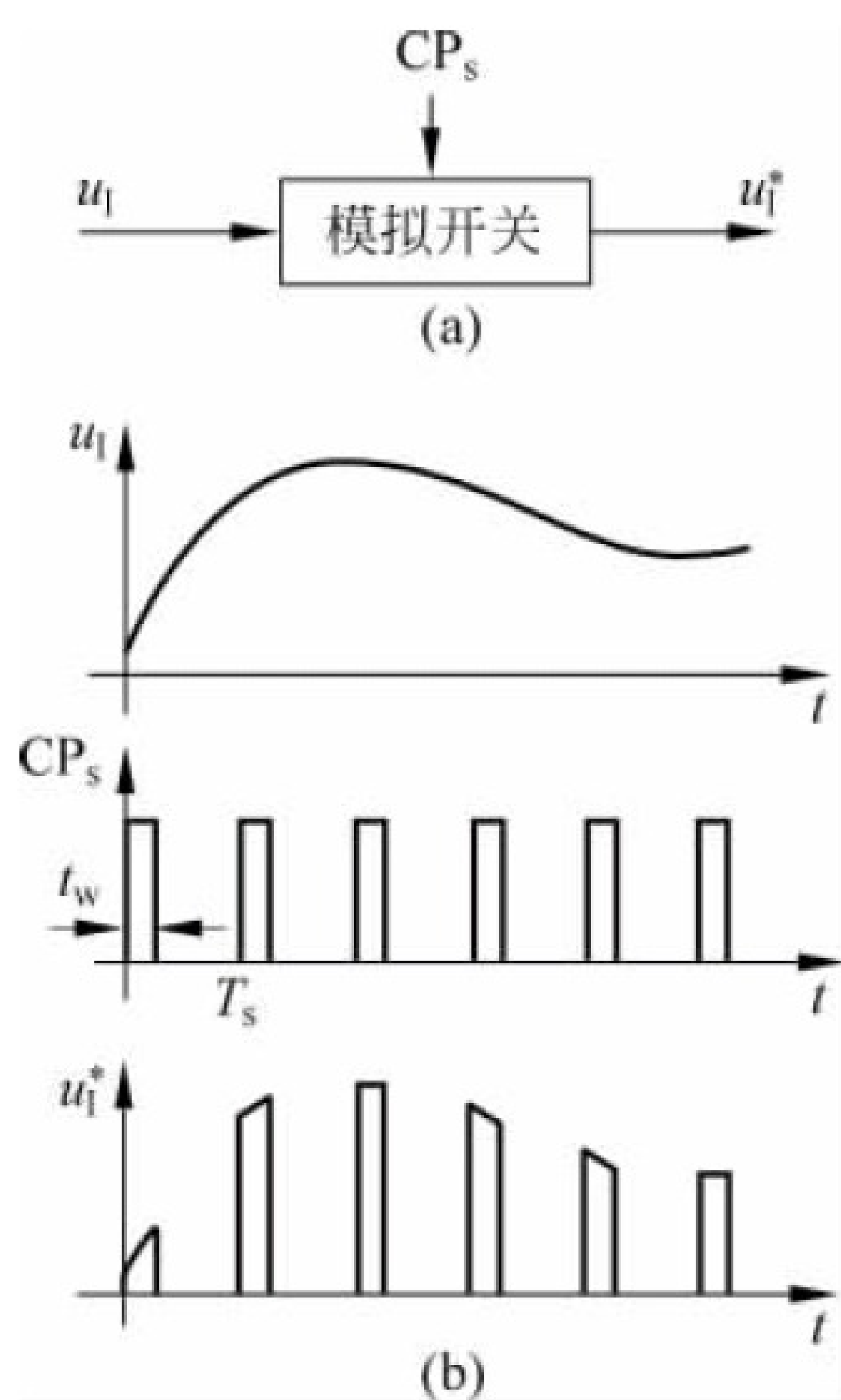


图 8.3.1　模拟信号的采样

为了能保证采样输出信号 u_I^* 能恢复成原信号，采样脉冲的频率 f_s应满足

$$f_s \geqslant 2f_{IM} \tag{8.3.1}$$

式中，f_{IM}为输入模拟量 u_I中的最高频率分量的频率。

式(8.3.1)称为采样定理。

由于每次将采样电压转换为相应的数字量都需要一定的时间，所以在每次采样以后，必须将采样电压 u_I^* 保持一段时间。由此可见，在进行 A/D 转换时所用的输入电压，实际上是每次采样结束时的 u_I值。

2. 采样—保持电路

如图 8.3.2(a)所示电路为常用的采样—保持电路。图中的 N 沟道 MOS 管 T_N作为采样开关用；C 作为保持信号用，故称为保持电容；运算放大器作为缓冲放大器用，它的输入阻抗越高，u_I^* 的保持时间越长。

当采样脉冲 CP_S到来，在 t_w时间内 T_N管导通，输入的模拟信号 u_I经 T_N管给电容 C 充电，假设电容 C 的充电时间常数远远小于 t_w，则电容 C 上的电压 u_C在时间 t_w内能跟随输入信号 u_I而变化，故运算放大器的输出 u_I^* 也随 u_I变化而变化，即 $u_I^*=u_C=u_I$。

当采样脉冲 CP_S结束时，MOS 管截止，但由于电容两端电压不能跳变，电路将保持采样信号一定的时间。由于运算放大器的输入阻抗很高，因此电容 C 的漏电与 A/D 转换的时间相比可以忽略。当下一采样脉冲到来时，u_I^* 又随 u_I的变化而变化。

图 8.3.2(b)所示电路的工作原理与图 8.3.2(a)所示电路的工作原理完全相同。只要满足时间常数 $R_2C \ll t_w$，输出电压 u_I^* 就会在采样时间内跟随 u_I 的变化而变化。图 8.3.2(b)电路比图 8.3.2(a)电路的输入阻抗稳定，但输出与输入反相。

实际电路中常使用集成采样—保持电路，它具有更好的性能。图 8.3.3 所示电路是单片集成采样—保持电路 LF198 的典型电路连接。图中，u_I 为模拟输入，u_L 为采样开关的输入控制；当 u_L 为 1 时，采样电路采样，$u_O = u_I$；当 u_L 为 0 时，电路将保持采样信号一定的时间。U_+、U_- 为运算放大器工作电源。U_B 为偏置电压输入，可通过调节 U_B 使 $u_I = 0$ 时 $u_O = 0$。

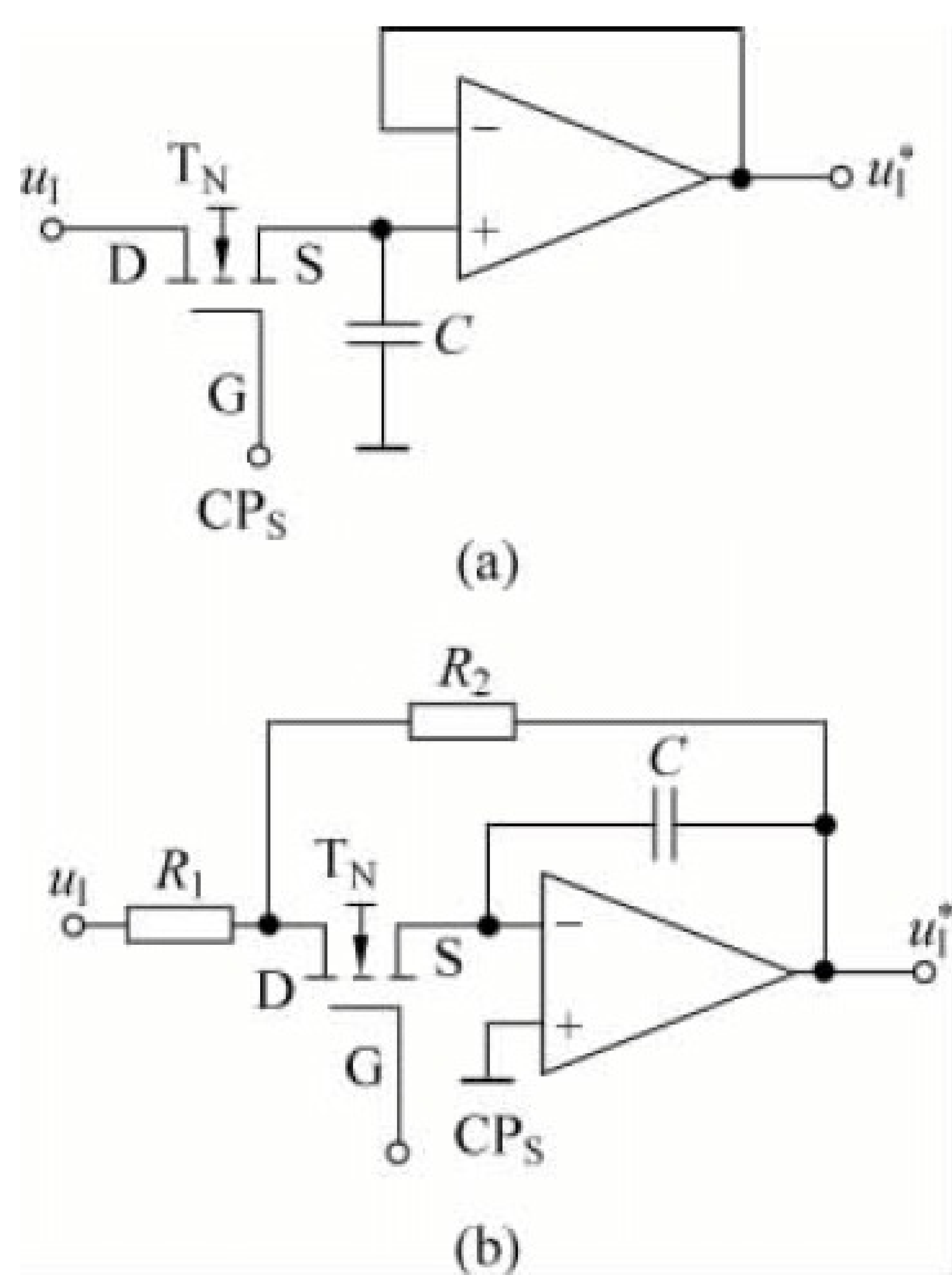


图 8.3.2　采样—保持电路

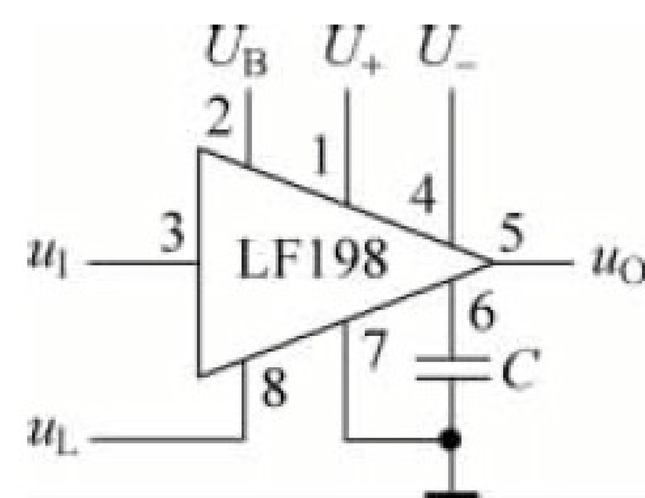


图 8.3.3　LF198 典型连接

3. 量化和编码

众所周知，数字信号不仅在时间上是离散的，而且在数值上的变化也是不连续的。任何一个数字量的大小，都是以某个最小数量单位的整数倍来表示的。在用数字量表示采样电压时，也必须将它化成这个最小数量单位的整数倍，这个转化过程称为量化。所规定的最小数量单位称为量化单位，用 Δ 表示。显然，数字信号最低有效位中的 1 所表示的数量大小，就等于 Δ。将量化的数值用二进制代码表示，称为编码，这个二进制代码就是 A/D 转换器输出的数字信号。

由于模拟电压是连续的，因此它不一定能被 Δ 整除，也就不可避免地会产生误差，这种误差称为量化误差。将模拟信号划分为不同的量化等级时，用不同的划分方法其量化误差也不相同。通常在划分量化等级时有两种方法：一种是只舍不入法，其量化单位 Δ 为

$$\Delta = \frac{U_{IM}}{2^n} \tag{8.3.2}$$

式中，U_{IM} 为输入模拟信号的最大值；n 为输出数字的位数。

另一种方法是四舍五入法量化方式，其量化单位 Δ 为

$$\Delta = \frac{2U_{IM}}{(2^{n+1}-1)} \tag{8.3.3}$$

图 8.3.4(a)所示为将 0～1V 的模拟电压信号转换成 3 位二进制代码($D_2D_1D_0$)的示意图。它是按照只舍不入的方式进行量化的。由式(8.3.2),其量化单位 Δ 为

$$\Delta = \frac{U_{IM}}{2^n} = \frac{1}{2^3} = \frac{1}{8}\text{V}$$

从图 8.3.4(a)中看出,凡数值为 0～1/8V 的模拟电压都当作 0×Δ 看待,用二进制数 000 表示,凡数值为 1/8～2/8V 的模拟电压都当作 1×Δ 看待,用二进制 001 表示等。不难看出,这种量化方式的最大量化误差为 Δ,即 1/8V。

(a)

模拟电平	二进制代码	二进制码代表的模拟电平
1V		
7/8V	111	7Δ=7/8V
6/8V	110	6Δ=6/8V
5/8V	101	5Δ=5/8V
4/8V	100	4Δ=4/8V
3/8V	011	3Δ=3/8V
2/8V	010	2Δ=2/8V
1/8V	001	1Δ=1/8V
0V	000	0Δ=0V

(b)

模拟电平	二进制代码	二进制码代表的模拟电平
1V		
13/15V	111	7Δ=14/15V
11/15V	110	6Δ=12/15V
9/15V	101	5Δ=10/15V
7/15V	100	4Δ=8/15V
5/15V	011	3Δ=6/15V
3/15V	010	2Δ=4/15V
1/15V	001	1Δ=2/15V
0V	000	0Δ=0V

图 8.3.4　电平量化的两种方式

图 8.3.4(b)所示是将 0～1V 的模拟电压信号转换成 3 位二进制代码($D_2D_1D_0$),按照四舍五入量化方式进行量化的示意图。其量化单位 Δ 为

$$\Delta = \frac{2U_{IM}}{(2^{n+1}-1)} = \frac{2}{(2^4-1)} = \frac{2}{15}\text{V}$$

从图 8.3.4(b)中看出,凡数值为 0～1/15V 的模拟电压都当作 0×Δ 看待,用二进制数 000 表示,凡数值为 1/15～3/15V 的模拟电压都当作 1×Δ 看待,用二进制数 001 表示等。不难看出,这种量化方式的最大量化误差为 1/2Δ,即 1/15V。

8.3.2　逐次逼近型模/数转换器

A/D 转换器一般分为两大类：一类是直接法；另一类是间接法。

所谓直接法是指,A/D 转换器将输入的模拟电压直接转换为输出的数字量,而不需要经过中间变量。常用的有并联比较型和反馈比较型两种。

这种转换方法的优点是工作速度快、调整方便,缺点是转换精度一般比间接法低。

所谓间接法是指,A/D 转换器将输入的模拟电压转换为中间变量(一般是时间 T 或者频率 f),然后再将中间量转换为数字量,常用的有电压—时间变换型和电压—频率变换型两种。

这种转换方法的特点是工作速度低、转换精度高、抗干扰能力强。

逐次逼近型 A/D 转换器属于直接法转换。其基本原理为，将大小不同的参考电压与采样—保持后的模拟电压 U_I逐次进行比较。把每次的比较结果用相应的二进制代码表示。其比较过程如同用天平称量一个未知质量的物体时的操作程序完全一样，只不过后者使用的砝码质量一个比一个小 1/2。

1. 电路组成

如图 8.3.5 所示为 3 位逐次逼近型 A/D 转换器的原理方框图。它由以下几部分组成。

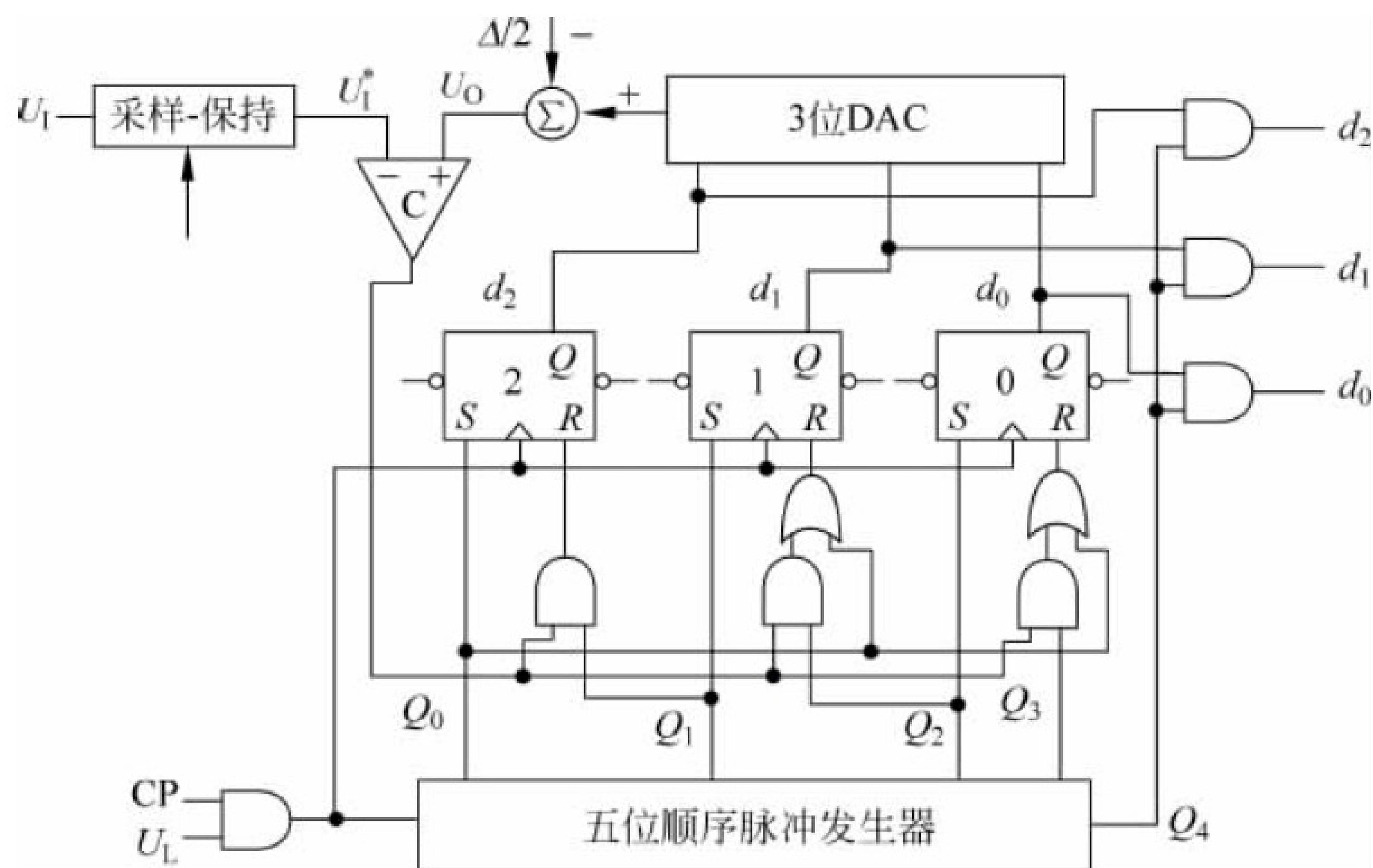


图 8.3.5　3 位逐次逼近型 A/D 转换器原理方框图

(1) D/A 转换器。根据不同的输入数码，产生一组数值不同的参考电压送到电压比较器与采样信号 U_I进行比较。D/A 转换器可以采用权电阻网络或倒 T 形解码网络的电路结构。

(2) 电压比较器。它将输入信号 U_I^* 与参考电压 U_O进行比较，若 $U_I^* > U_O$时输出 $U_C = 0$；若 $U_I^* < U_O$时输出 $U_C = 1$。

(3) 5 位顺序脉冲发生器。它的作用是产生比较用的五位顺序脉冲(也称为节拍脉冲)，可由模为 5 的环形计数器组成。其初值($Q_4Q_3Q_2Q_1Q_0$)为 00001，在 CP 脉冲作用下进行右移，在每一个输出端轮流出现 1，为模值为 5 的环形计数器，读者可结合图 6.4.67(4 位节拍脉冲仿真时序图)理解 5 位顺序脉冲发生器。

(4) 逐次逼近寄存器。它由 RS 触发器组成，在节拍脉冲作用下记下每次比较结果，并向 D/A 转换器提供输入数码。

2. 逐次逼近原理

逐次逼近类似计算机程序中的折半查找，可通过图 8.3.6 所示的区间分割原理来理解逐次逼近。图中假定输入数据 x 位于数据轴[a、b]之间，假定区间[a、b]可进一步划分为 8 个子区间(用数字 0～7 表示)，则可通过下面的方法判断输入 x 位于区间[a、b]的哪一个子区间：

(1) 先取区间[a、b]中值(子区间4)与输入 x 比较,可判断 x 属于区间[a、b]的右半区。

(2) 取区间[a、b]右半区的中值(子区间6)与输入 x 比较,可判断 x 属于区间[a、b]右半区的左半区。

(3) 取区间[a、b]右半区的左半区的中值(子区间5)与输入 x 比较,可判断 x 属于子区间5。

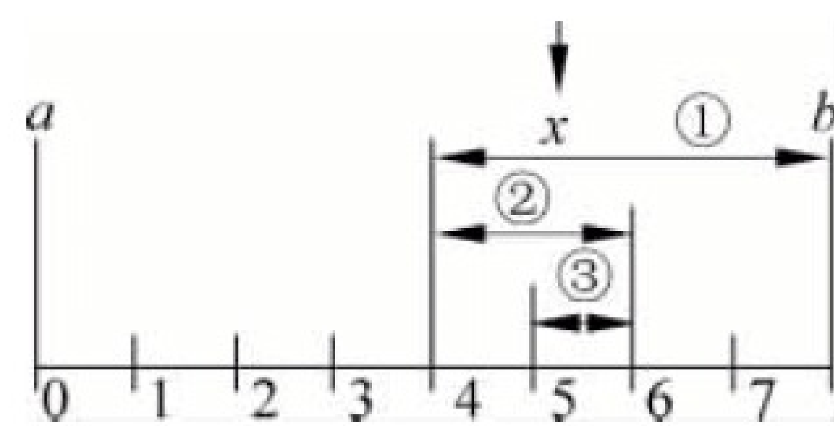


图8.3.6 逐次逼近原理

上述过程便是A/D转换中的逐次逼近原理。有兴趣的读者可结合下面内容进一步理解上面逐次逼近原理的电路实现方法。

设输入信号为4.9V,量化单位Δ为1V,令转换控制信号 $U_L=1$。转换前初态($Q_0Q_1Q_2Q_3Q_4$)为10000、逐次逼近寄存器输出 $d_2d_1d_0$ 为000。

① 当第1个CP脉冲到来时,$Q_0=1$($Q_1\sim Q_4$ 为0),逐次逼近寄存器输出 $d_2d_1d_0=100$(取区间中值)。该数据送给3位D/A转换器,得到参考电压 $U_O=4-0.5=3.5$V(为保证转换精度,图中D/A转换器输出产生了$-\Delta/2$的偏移。

电压比较器C将参考电压 U_O 与采样系统保持的瞬间输入信号 U_I^* 进行比较,当 $U_I^*>U_O$ 时输出 $U_C=0$;当 $U_I^*<U_O$ 时输出 $U_C=1$。由于 $U_I^*=4.9$V,故输出 $U_C=0$。

由于CP脉冲的作用,5位顺序脉冲发生器右移,输出($Q_0Q_1Q_2Q_3Q_4$)为01000。

② 当第2个CP脉冲到来时,$Q_1=1$(其余 Q 均为0),逐次逼近寄存器输出应根据上一次比较结果选择左半区或右半区中值作为进一步比较的参考。

$Q_1=1$ 将使 $d_1=1$(取左半区或右半区的中值),故可通过是否保留 d_2 的"1"来确定是选择左半区还是右半区。显然,$U_C=1$,2号RS触发器 R 端为1,2号RS触发器清零;$U_C=0$,2号RS触发器 R 端为0,RS触发器状态不变。即当 $U_I^*>U_O$ 时,选择右半区;反之,选择左半区。由于 $U_C=0$,故选择右半区,$d_2d_1d_0=110$。经3位D/A转换器得到参考电压 $U_O=6-0.5=5.5$V。由于 $U_I^*=4.9$V,故输出 $U_C=1$。

5位顺序脉冲发生器继续右移,输出($Q_0Q_1Q_2Q_3Q_4$)为00100。

③ 当第3个CP脉冲到来时,$Q_2=1$(其余 Q 均为0),逐次逼近寄存器输出应根据上一次比较结果选择待比较区间的中值作为进一步比较的参考。

$Q_2=1$ 将使 $d_0=1$(待比较区间的中值),故可通过是否保留 d_1 的"1"来确定是选择上一次比较区间的左半区还是右半区(由于2号RS触发器 R 端为0,RS触发器状态不变,故 d_2 的值不变)。由于 $U_C=1$,故选择左半区,$d_2d_1d_0=101$。经3位D/A转换器得到参考电压 $U_O=5-0.5=4.5$V。由于 $U_I^*=4.9$V,故输出 $U_C=0$。

5位顺序脉冲发生器继续右移,输出($Q_0Q_1Q_2Q_3Q_4$)为00010。

④ 当第4个CP脉冲到来时,$Q_3=1$(其余 Q 均为0),逐次逼近寄存器将确定选择上一次比较区间的左半区还是右半区作为最终结果。

由于 $U_C=1$,故选择右半区,$d_2d_1d_0=101$,为最终结果。

5位顺序脉冲发生器继续右移,输出($Q_0Q_1Q_2Q_3Q_4$)为00001。

⑤ 当第 5 个 CP 脉冲到来时，$Q_4=1$(其余 Q 均为 0)，$d_2d_1d_0$保持不变，同时，由于 $Q_4=1$，将最终结果输出。

5 位顺序脉冲发生器继续右移，输出($Q_0Q_1Q_2Q_3Q_4$)为 10000。

综上所述，从逐次逼近过程可以看出，三位输出的 A/D 转换器完成一次转换需要 5 个 CP 脉冲信号的周期，可以证明，n 位输出的 A/D 转换器完成一次转换需要 $n+2$ 个 CP 脉冲信号的周期，故转换速度不够高。但逐次逼近型 A/D 转换器的分辨率高，转换误差较低，是目前应用比较广泛的一种 A/D 转换器。

8.3.3　其他常用模/数转换器



1. 并联比较型 A/D 转换器

并联比较型 A/D 转换器属于直接型 A/D 转换器，将输入的模拟电压直接转换为输出的数字量。如图 8.3.7 所示为 3 位并联比较型 A/D 转换器原理图，包括电压比较器、代码转换器、寄存器 3 个部分。

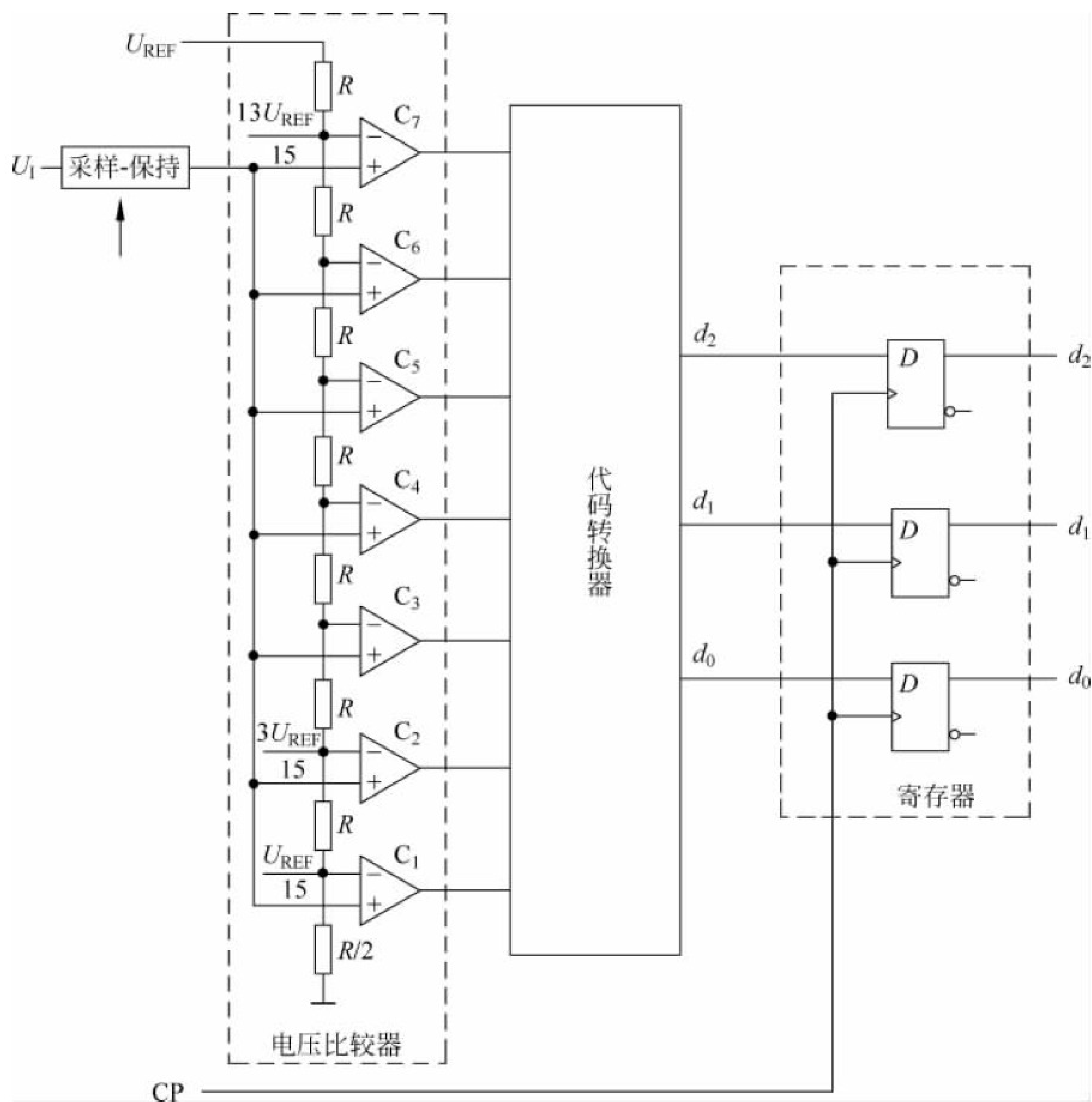


图 8.3.7　3 位并联比较型 A/D 转换器原理图

依照如图 8.3.4(b)所示的四舍五入的量化方法,选择量化单位 Δ 为 $\frac{2}{15}U_{\text{REF}}$。电阻分压器产生 $\frac{1}{15}U_{\text{REF}} \sim \frac{13}{15}U_{\text{REF}}$ 之间的 7 个比较电平,并将这 7 个比较电平作为比较基准分别接到 7 个比较器的输入端。同时,将输入电压加到每个比较器的另一个输入端,与这 7 个基准电平进行比较,若大于某个基准电平则输出为 1,反之为 0。

输入电压与 7 个基准电平进行比较,比较结果可从大于 0 个基准电平到大于 7 个基准电平等 8 个状态变化,相应地将获得 8 个输出代码。每个输出代码具有 7 个二进制位,对应转换结果中的一个 3 位二进制数。可用一个 7 输入、3 输出的组合逻辑电路来实现电压比较器直接比较结果到 3 位转换结果的转换。有兴趣的读者可根据该逻辑问题的特点,自行设计该组合逻辑电路。

转换完成后,在时钟信号 CP 的作用下,将 3 位二进制数存入寄存器中后结束本次转换。

并联型 A/D 转换器最大优点是转换速度快,因此也称闪速 A/D 转换器,如 8 位 A/D 转换器 ADC08100 的转换时间仅为 10ns;主要缺点是电路复杂,n 位的 A/D 转换器中,需要使用 2^{n-1} 个电压比较器,相应的代码转换电路输入也有 2^{n-1} 个。

2. 半闪速 A/D 转换器

为了在增加输出位数的同时不使电路的规模过于膨胀,可将并联型 A/D 转换器和反馈比较型 A/D 转换器二者结合,采用“半闪速”的电路结构。

如图 8.3.8 所示为 8 位半闪速 A/D 转换器原理图,包括两个 4 位的并联型闪速 A/D 转换器和一个 4 位的 D/A 转换器。

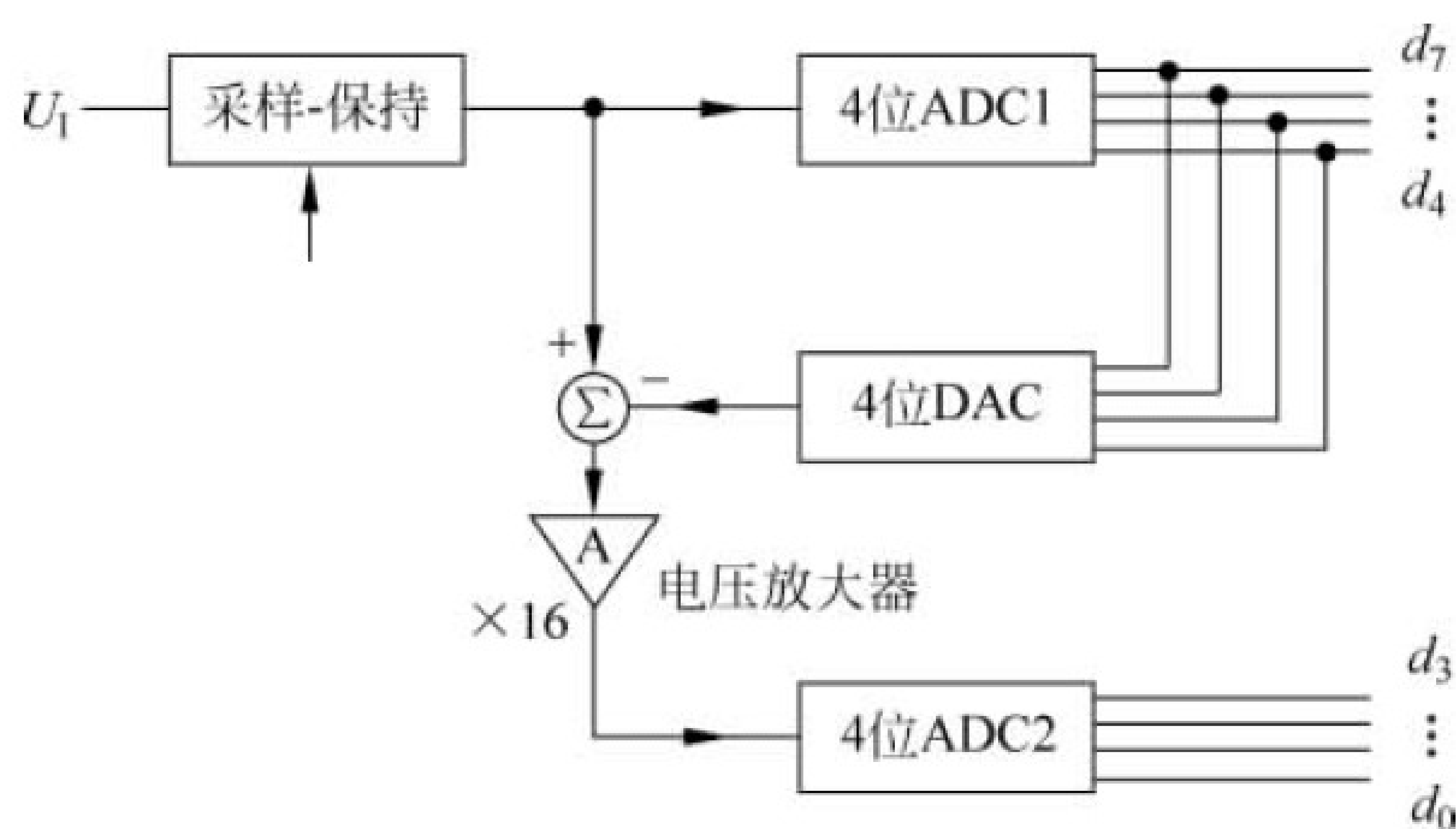


图 8.3.8 8 位半闪速 A/D 转换器

输入电压先与高 4 位的并联型闪速 A/D 转换器 ADC1 的 15 个基准电平进行比较,相应地获得一个 4 位的二进制数,对应转换结果的高 4 位二进制数。将获得的 4 位的二进制数反馈给 4 位的 DAC,输出相应的模拟电压值。

将输入电压减去 4 位的 DAC 输出相应的模拟电压值,可获得相应的残余电压用于低 4

位二进制数的转换。

显然，对残余电压进行量化的单位为高 4 位 ADC1 的量化单位的 1/16，相应的转换参考电压也为 ADC1 的转换参考电压的 1/16。为保证 ADC1、ADC2 两个转换器使用相同的转换参考电压，避免 ADC2 工作在微弱输入信号的工作状态，可将残余电压用电压放大器放大 16 倍后送 ADC2。

显然，ADC2 需要等 ADC1 转换完成且 4 位 DAC 也转换完成后才可求出正确的转换结果，转换速度将显著地慢于纯粹的并联型 A/D 转换器。尽管如此，半闪速 A/D 转换器的转换速度依然是比较快的，如半闪速 A/D 转换器 ADC0820 的转换时间仅为 1.5μs。此外，半闪速 A/D 转换器大大降低了电路的复杂程度。如图 8.3.8 所示的 8 位半闪速 A/D 转换器的每个 4 位 ADC 只需要 15 个电压比较器，共需要 30 个电压比较器，而 8 位纯粹的并联型 A/D 转换器需要 255 个电压比较器。

其他常用的 A/D 转换器还有双积分型 A/D 转换器、V-F 变换型 A/D 转换器等。

双积分型 A/D 转换器属于电压—时间变换型 A/D 转换器，其基本原理是通过积分器将模拟电压信号变换为与之成正比的时间宽度信号，然后在这个固定的时间宽度里对固定频率的时钟脉冲计数，从而获得最终结果。

双积分型 A/D 转换器的主要优点是具有很强的抗干扰能力，电路比较简单，便于集成，容易提高分辨率；主要缺点是需要双积分过程，所以转换速度较低，一般为几毫秒到几十毫秒。

V-F 变换型 A/D 转换器为电压—频率变换型 A/D 转换器，其基本原理是通过压控振荡器(VCO)将模拟电压信号变换为与之成正比的频率信号，然后在一个固定的时间间隔里对得到的频率信号计数，从而获得最终结果。

V-F 变换型 A/D 转换器的主要优点是 VCO 的输出信号是一个频率信号，便于传输和检出，非常适合于在遥测、遥控系统中应用，也具有很强的抗干扰能力。其主要缺点其精度取决于 VCO 的线性度和稳定度，一般的 V/F 型 A/D 转换器的线性误差均较大；另一个缺点是转换速度也较低。

对于双积分型 A/D 转换器、V-F 变换型 A/D 转换器，有兴趣的读者可参考相关书籍。

8.3.4 模/数转换器的实例



A/D 转换器集成芯片种类较多，下面以 ADC0809 为例介绍其功能及应用。

图 8.3.9 所示为 ADC0809 的引脚图，为 8 路输入 3 位逐次逼近型 A/D 转换器。其转换原理与图 8.3.5 所示 3 位逐次逼近型 A/D 转换器转换原理相同。另外，它从方便应用角度增加了模拟输入通路选择，输出锁存及控制等。

各引脚功能如下：

$IN_0 \sim IN_7$：8 路模拟量输入端。

A,B,C：8 路模拟量输入选择控制端，按(CBA)排列顺序选择对应的模拟输入量，如

$(CBA)=(001)_2$选择 IN_1 作为输入进行转换。

ALE：地址锁存输入端，高电平有效，可加正脉冲。

$D_0 \sim D_7$：8 路数字量输出端。

EOC：转换结束输出端，高电平有效。

EOUT：输出允许端，高电平有效。

START：转换启动信号输入端，可加正脉冲，上升沿转换器清零，下降沿开始转换。

CP：外部时钟输入端，典型频率为 640kHz。

$U_{REF(-)}$，$U_{REF(+)}$：转换器参考电源输入端。

U_{DD}，GND：转换器工作电源，电压为＋5V。

ADC0809 的应用实例如图 8.3.10 所示。实现 A/D 转换的过程如下：

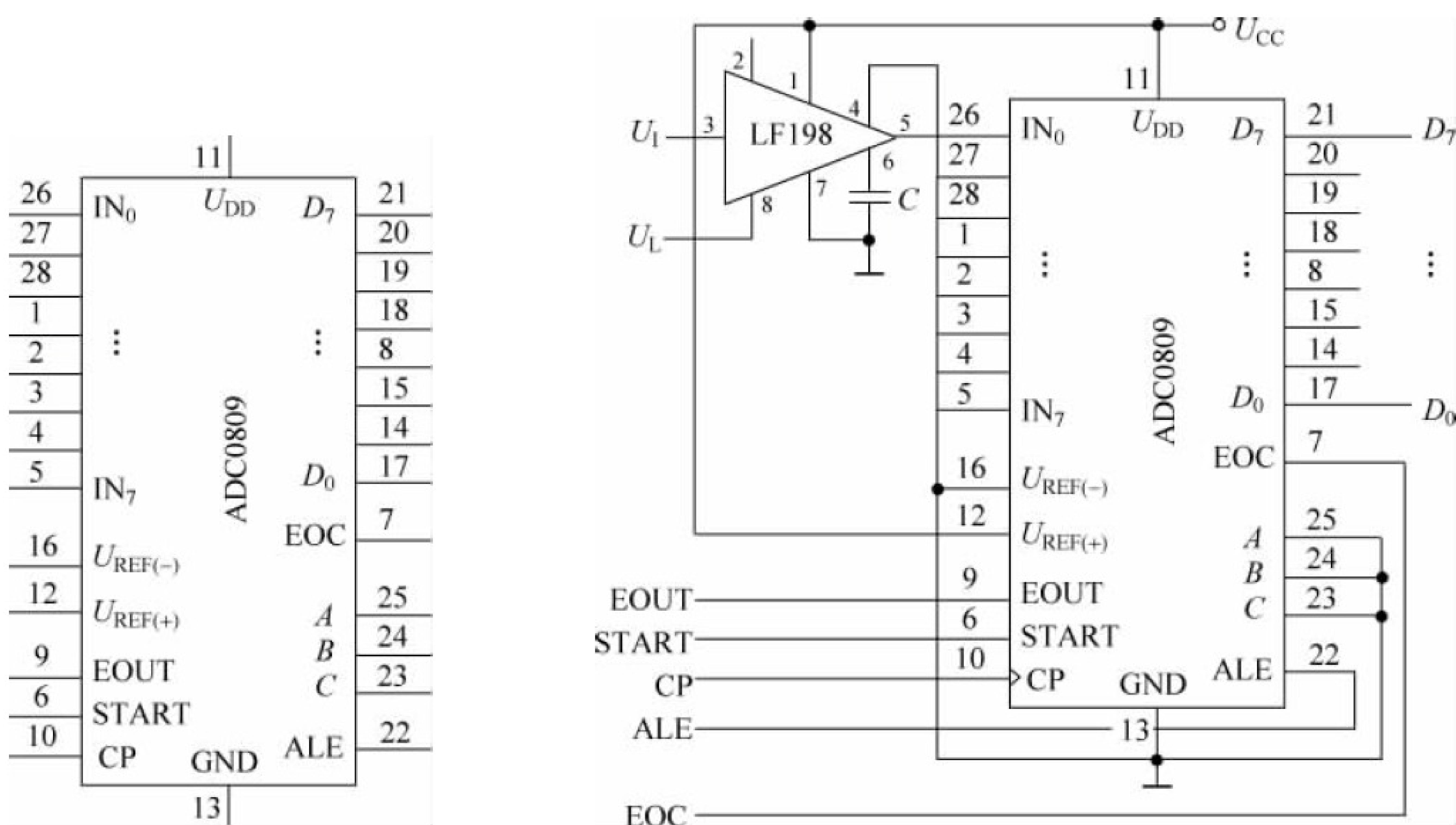


图 8.3.9　ADC0809　　　　图 8.3.10　ADC0809 应用实例

（1）在 U_L 端加一个具有一定时间宽度的正脉冲（$U_L=1$，电路对输入信号采样；待采样信号稳定后，$U_L=0$，LF198 将对采样信号保持一段时间供 A/D 转换）。

（2）在 ALE 端加一个正脉冲，选择 IN_0 进入 A/D 转换器。

（3）延时一个时钟信号。

（4）在 START 端加一个正脉冲，启动 A/D 转换器对 IN_0 进行转换。

（5）反复查询 EOC 状态，直到 EOC＝1。

（6）令 EOUT＝1，读取 $D_0 \sim D_7$。

8.3.5　模/数转换器的转换精度与转换速度

1. 转换精度

在 A/D 转换器中，用分辨率和转换误差描述转换精度。

(1) 分辨率。分辨率以输出二进制或十进制数字的位数来表示，它说明 A/D 转换器对输入信号的分辨能力。输出为 n 位二进制数的 A/D 转换器能区分输入模拟电压信号的 2^n 个不同等级。

(2) 转换误差。转换误差以相对误差来表示。它表示模数转换器实际输出的数字量和理想输出数字量之间的差别，并用最低有效位的倍数来表示。例如，给出相对误差小于或等于 1/2LSB，表明实际输出的数字量和理论上应得到的输出数字量之间的误差不大于最低位的 1/2。

2. 转换速度

A/D 转换器的转换速度主要取决于转换电路的类型，不同类型的 A/D 转换器速度相差很大。

并联型 A/D 转换器的转换速度最高，8 位输出单片集成 A/D 转换器的转换时间一般不超过 50ns。逐次逼近型 A/D 的转换器次之，8 位输出单片集成 A/D 转换器的转换时间一般为 10～50μs。间接转换器的转换速度要低得多，例如双积分型单片集成模数转换器的转换时间，大都为几十毫秒至几百毫秒。

复习与思考

8.3.1　在如图 8.3.5 所示 A/D 转换器中，设输入信号为 3.7V，量化单位 Δ 为 1V，说明其 A/D 转换过程及最终转换结果。

8.3.2　如图 8.3.5 所示 A/D 转换器的最小可分辨电压为多少(量化单位 Δ 为 1V)?

8.3.3　利用你所学的知识设计如图 8.3.7 所示 3 位并联比较型 A/D 转换器中的代码转换电路。

习题

8-1　填空题

1. 常见的非电物理量可通过相应的传感器等变换为随时间连续变化的电信号，称为________。如果要将这些________送到数字系统中处理，必须先将这些________转换成数字量，实现这一功能的电路称为________，简称________或________。

2. 可以将 D/A 转换器看成是一个________，它是将输入的二进制数字信号 D(或称为编码信号)转换(翻译)成模拟信号，并以________的形式输出。D/A 转换器所能分辨出的最小输出电压与最大输出电压之比称为________，n 位 D/A 转换器分辨率为________。

3. 在用数字量表示采样电压时，也必须将它化成这个________单位的整数倍，这个转化过程称为________。所规定的________单位称为________，用 Δ 表示。将________的数值用二进制代码表示，称为________。

4. ________ A/D 转换器最大优点是转换速度快，也称________ A/D 转换器，主要缺点是电路复杂。n 位的 A/D 转换器中，需要使用________个电压比较器，相应的代码转换电路输入也有________个，可将________ A/D 转换器和反馈比较型 A/D 转换器二者结合，这种电路结构称为________的电路结构。

8-2 分析应用题

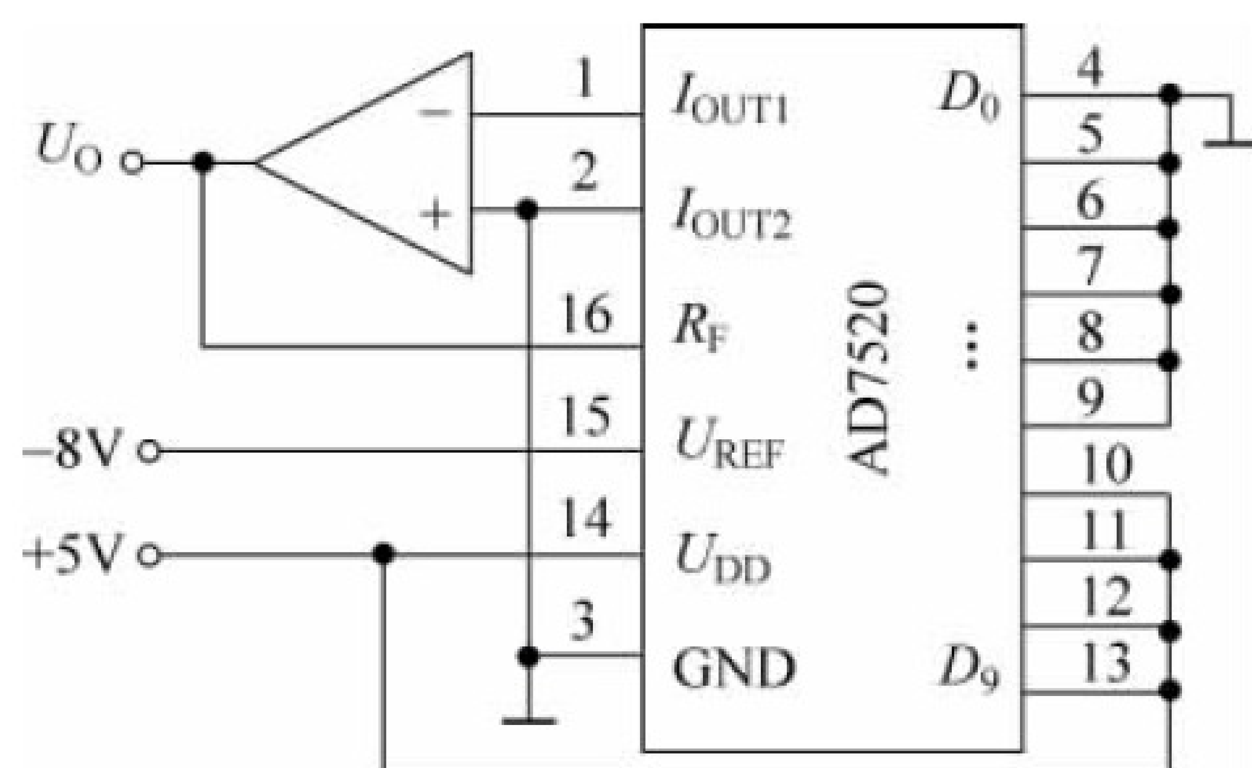
1. 在如图 8.2.5 所示的倒 T 形电阻网络 D/A，已知 $R=R_F$，$U_{REF}=-8V$，试计算输入数字量从全 0 变到全 1 时，输出电压的变化范围。

2. 已知 D/A 转换器的最小分辨电压 $U_{LSB}=2.442mV$，最大满刻度输出模拟电压 $U_{OM}=10V$，求该转换器输入二进制数字量的位数 n。

3. 在 10 位二进制数的 D/A 中，已知最大满刻度输出模拟电压 $U_{OM}=5V$，求最小分辨电压 U_{LSB}和分辨率。

4. 已知某 D/A 电路的输入二进制数字量的位数 $n=9$，最大满刻度输出模拟电压 $U_{OM}=5V$，试求最小分辨电压 U_{LSB}、分辨率和参考电压 U_{REF}。

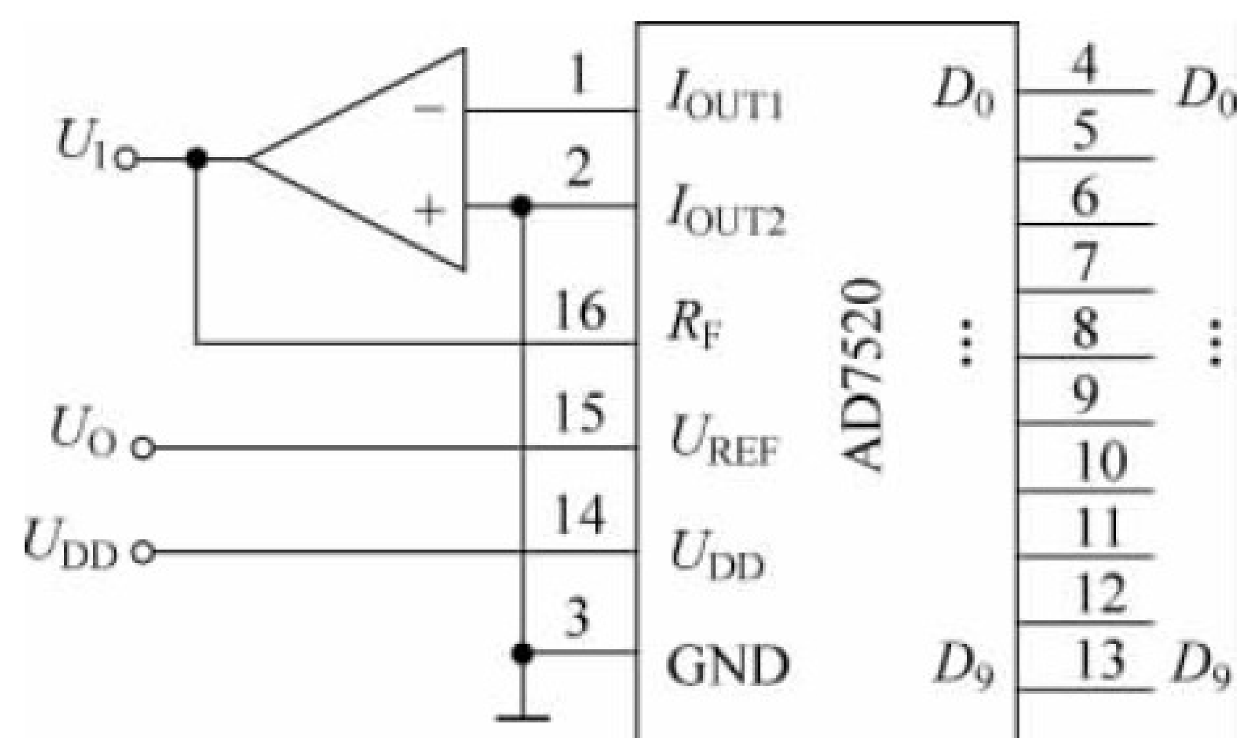
5. 求如题图 8.1 所示电路中 U_O的值。



题图 8.1

6. 如图 8.2.8 所示权电容网络 D/A 转换器中的参考电压 $U_{REF}=10V$，输入的数字量 $D_3D_2D_1D_0=1000$，求电路中 U_O的值。

7. 如题图 8.2 所示电路中，输入电压 $U_I=20mV$，输入的 10 位二进制数 10 进制值为 32，求电路中 U_O的值。



题图 8.2

8. 在 A/D 转换过程中，量化有哪两种方式？它们各自产生的量化误差是多少？试举例说明。

9. 已知输入电压 $U_I=0\sim10V$；对于 $n=4$ 的逐次逼近 A/D 转换器电路，用四舍五入方式量化，求 $U_I=6.28V$ 时，输出的数字量 $D_3D_2D_1D_0$。

10. 已知输入电压 $U_I=0\sim8V$；对于 $n=4$ 的逐次逼近 A/D 转换器电路，用四舍五入方式量化，求 $U_I=5.28V$ 时，输出的数字量 $D_3D_2D_1D_0$。

11. 在逐次逼近 A/D 转换器中，若要求输出 8 位二进制数码，说明输入电压为 20.5V 时，输出 8 位二进制数码是什么？假设 D/A 转换器输出电压的阶梯是 0.1V。

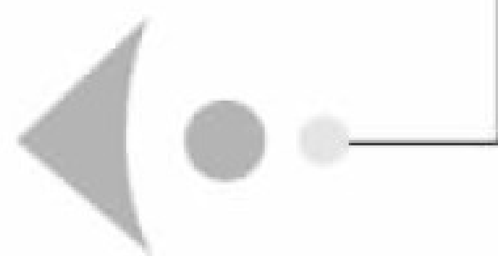
12. 试比较直接式 A/D 转换和间接式 A/D 转换的优缺点。

13. 在如图 8.3.10 所示电路中，假定 $U_I=3.35V$，工作电源为 $+5V$，假定 ADC0809 采用只舍不入法进行转换，求最终转换结果。

14. D/A 转换器和 A/D 转换器的转换精度及转换速度与哪些因素有关？



第9章



脉冲单元电路

本章要点：

脉冲电路是绝大多数数字系统的必备电路。学习本章应理解脉冲信号的基本参数；理解施密特触发器、单稳态触发器、多谐振荡器等脉冲电路的特点；在此基础上，掌握利用555定时器构成施密特触发器、单稳态触发器、多谐振荡器等脉冲电路的方法。

9.1 概述



脉冲信号是一种持续时间极短的电压或电流波形，如数字系统的时钟信号。时钟信号是一种矩形脉冲信号，是时序逻辑电路的基本工作信号，它可通过脉冲单元电路来获得。

前面各章涉及的矩形脉冲为理想脉冲，实际的矩形脉冲波形如图9.1.1所示。下面介绍与矩形脉冲波形相关的主要参数。

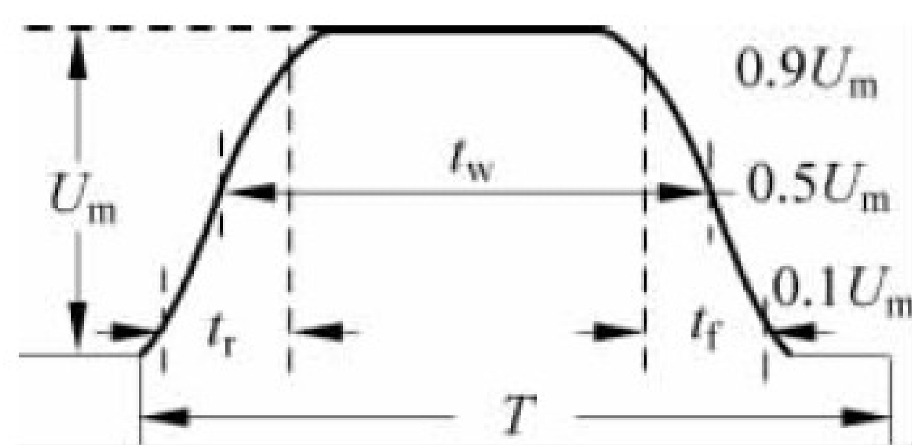


图9.1.1　矩形波的参数

脉冲幅度 U_m：脉冲电压的最大变化量称为脉冲幅度。

脉冲周期 T：周期性的重复脉冲，其两个相邻脉冲之间的时间间隔，称为脉冲的周期 T。周期 T 的倒数 $f=1/T$，称为脉冲的频率。

上升沿时间 t_r：脉冲电压幅度从 $0.1U_m$ 上升到 $0.9U_m$ 所需的时间称为脉冲的前沿上升时间(上升沿时间)。

下降沿时间 t_f：脉冲电压幅度从 $0.9U_m$ 下降到 $0.1U_m$ 所需的时间称为脉冲的后沿下降时间(下降沿时间)。

脉冲宽度 t_w：从脉冲前沿上升到 $0.5U_m$ 处开始，到脉冲后沿下降到 $0.5U_m$ 处为止的一段时间，称为脉冲宽度。

占空比 q：脉冲宽度 t_w 与脉冲周期 T 的比值，即 $q=t_w/T$。

上述各参数含义可通过图 9.1.1 来进一步理解。利用这些参数,可以表示矩形脉冲的基本特性。

显然,对于理想的矩形脉冲,其上升沿时间和下降沿时间均为零。

获取矩形脉冲的方式主要有两种:一种是利用各种形式的多谐振荡器电路直接产生需要的矩形脉冲;另一种是利用各种整形电路把已有的非矩形脉冲周期信号变换为符合要求的矩形脉冲。

脉冲电路是用来产生和处理脉冲信号的电路。时钟信号是时序电路的工作信号,产生时钟的脉冲电路是绝大多数数字系统的必备电路。常用的脉冲波形产生、整形电路有施密特触发器、单稳态触发器、多谐振荡器等。必须指出,脉冲电路和数字电路是两种性质完全不同的电路,脉冲电路中的触发器和数字电路中的触发器也是两个完全不同的概念,通过后面的学习将看到二者之间存在着本质的区别。

9.2 施密特触发器

施密特触发器是脉冲波形变换中经常使用的一种电路,应用十分广泛。

9.2.1 什么是施密特触发器



施密特触发器逻辑符号如图 9.2.1 所示。如图 9.2.1(b)所示施密特触发器也称施密特反相器。施密特触发器的电压传输特性如图 9.2.2 所示。

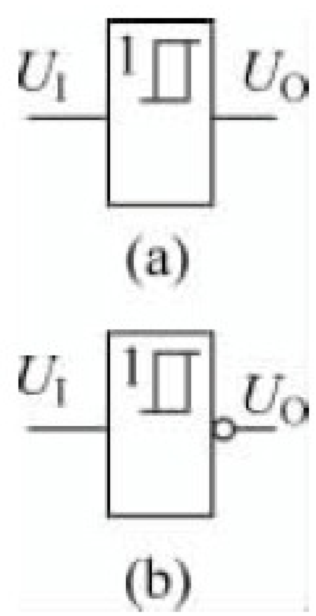


图 9.2.1 逻辑符号

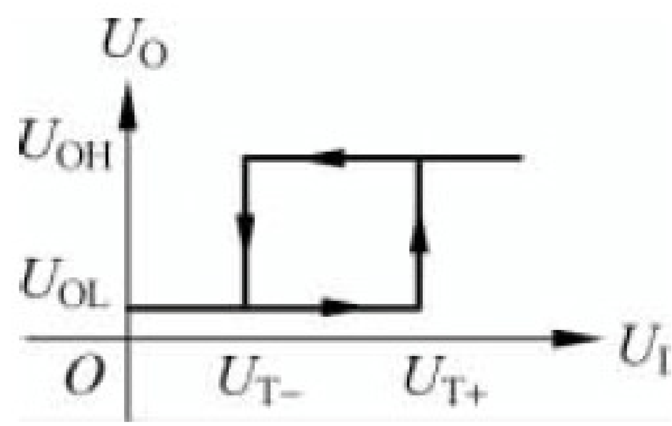


图 9.2.2 传输特性

由如图 9.2.2 所示电压传输特性可看出,施密特触发器具有两个稳态,而且每个稳态都需要外加信号才能维持,一旦输入信号撤除后,稳态会自动消失。此外,电压传输特性还有两个特殊的值:U_{T+} 和 U_{T-}。把输入信号 U_I(电路中,习惯上直流量用大写字母表示,交流量用小写字母表示,本书脉冲电路单元统一用大写字母表示各信号)从低电平上升的过程中,电路由一个稳态转换到另一个稳态所对应的输入电平称为接通电位,记为 U_{T+}。输入信号从高电平的下降过程中电路由一个稳态转换到另一个稳态所对应的输入电平称为断开电位,记为 U_{T-}。

施密特触发器在性能上具有两个重要的特点:

(1) 接通电位 U_{T+} 和断开电位 U_{T-} 不相同，满足 $U_{T+} > U_{T-}$。

定义 U_{T+}、U_{T-} 的差为回差电压 ΔU_T。回差电压是施密特触发器的重要参数，有

$$\Delta U_T = U_{T+} - U_{T-} \tag{9.2.1}$$

(2) 施密特触发器可以将变化非常缓慢的输入脉冲波形，整形成为适合于数字电路所需要的矩形脉冲。

综上所述，施密特触发器并不是第5章中介绍的触发器，二者是性质完全不同的两种电路。施密特触发器没有存储功能，输出端的逻辑状态随输入端的状态改变，是脉冲电路的一种。施密特触发器的英文名称(Schmitt Trigger)和第5章中介绍的触发器(Flip-Flop)的英文名称原本也截然不同，但是最初将 Schmitt Trigger 译成中文时用了“施密特触发器”这个名称并一直被沿用。读者切记不可将二者混淆。

9.2.2 用门电路组成的施密特触发器



如图9.2.3所示为用 CMOS 门电路组成的施密特触发器。工作原理简要分析如下：

CMOS 电路标准低电平为0V，标准高电平为电源值 U_{DD}，输入电流近似为0mA，阈值电平约为 $1/2U_{DD}$。依照 CMOS 电路的特点，如图9.2.3所示电路具有两个稳态，且每个稳态需要相应的输入维持。

当输入 $U_I=0$ 时，$U_{O1}=1$，$U_O=0$，电路处于稳态。

类似地，当输入 $U_I=1$ 时，$U_{O1}=0$，$U_O=1$，电路也处于稳态。

下面计算如图9.2.3所示电路的接通电位 U_{T+}。

$U_I=0$ 时，$U_O=0$，因此，$U_A=0(0V)$。

当 U_I 从低电平上升时，U_O 将暂时保持0V不变。因为 $U_I > U_O$，电流 i 实际方向如图9.2.4所示，U_A 电位将上升。

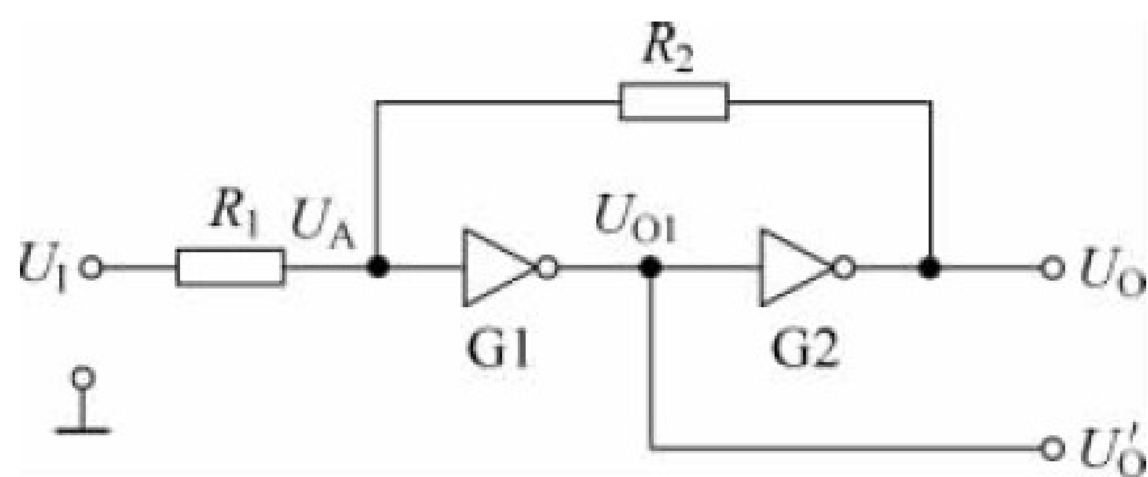


图9.2.3 用 CMOS 门电路组成的施密特触发器

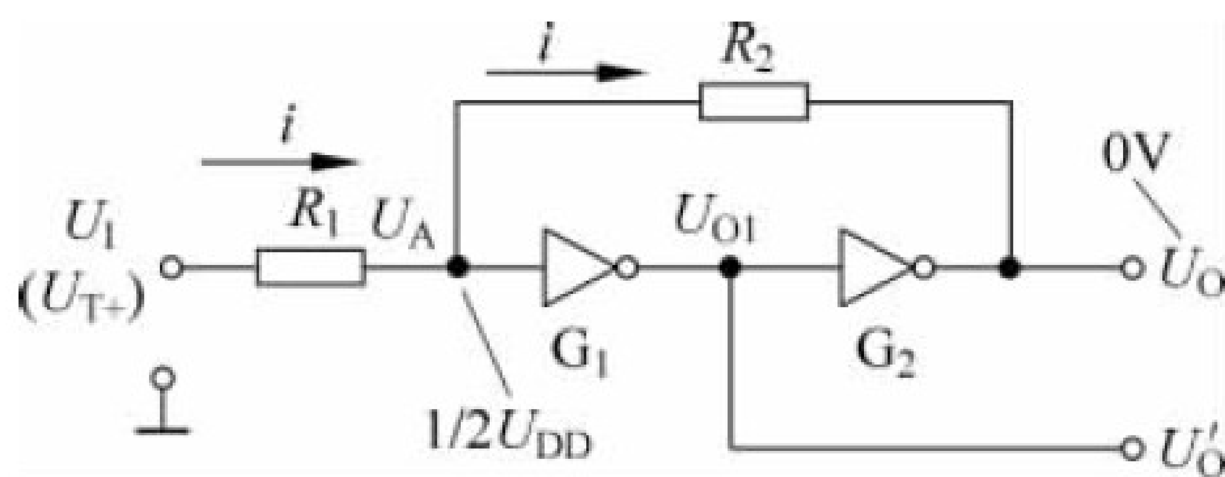


图9.2.4 接通电位 U_{T+}

当 U_A 电位上升到阈值电平($1/2U_{DD}$)后，门 G1 输出 U_{O1} 将翻转到0，$U_O=1$，在即将翻转的瞬间，U_I、U_O、U_A 实际电位如图9.2.4所示。有

$$U_{T+} = U_I = iR_1 + U_A = \frac{U_A - U_O}{R_2}R_1 + U_A$$

$$= \frac{1}{2}U_{DD}\frac{R_1}{R_2} + \frac{1}{2}U_{DD} = \left(1 + \frac{R_1}{R_2}\right)\frac{1}{2}U_{DD} \tag{9.2.2}$$

继续计算如图 9.2.3 所示电路的断开电位 U_{T-}。

$U_I=1$ 时，$U_O=1$，因此，$U_A=1(U_{DD})$。

当 U_I从高电平往下降时，U_O将暂时保持 $1(U_{DD})$不变。因为 $U_O>U_I$，电流 i 的实际方向如图 9.2.5 所示，U_A电位将下降。

当 U_A电位下降到阈值电平$(1/2U_{DD})$后，门 G1 输出 U_{O1}将翻转到 1，$U_O=0$，在即将翻转的瞬间，U_I、U_O、U_A实际电位如图 9.2.5 所示。有

$$U_{T-}=U_I=U_A-iR_1=U_A-\frac{U_O-U_A}{R_2}R_1$$

$$=\frac{1}{2}U_{DD}-\frac{1}{2}U_{DD}\frac{R_1}{R_2}=\left(1-\frac{R_1}{R_2}\right)\frac{1}{2}U_{DD} \tag{9.2.3}$$

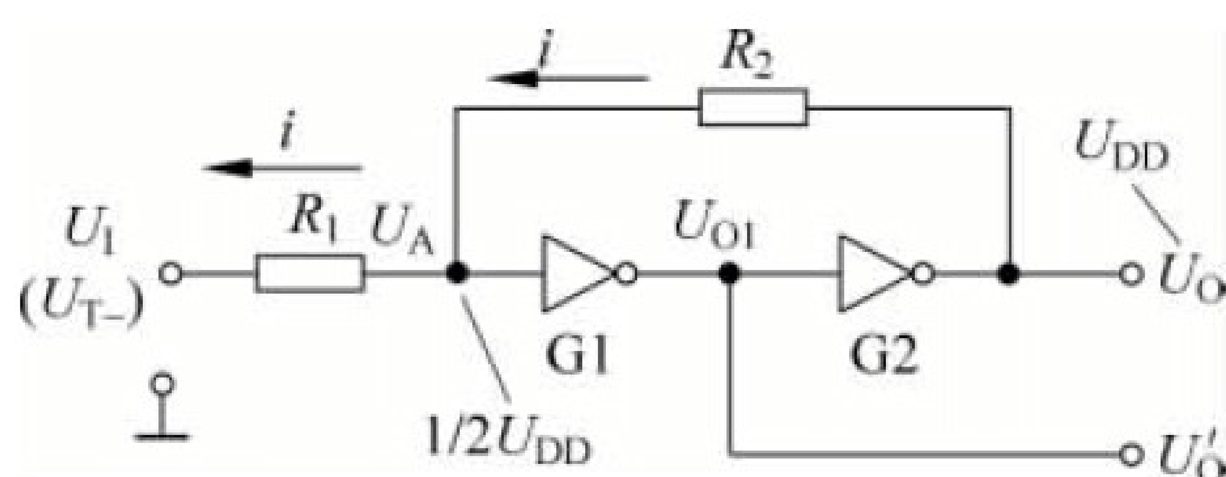
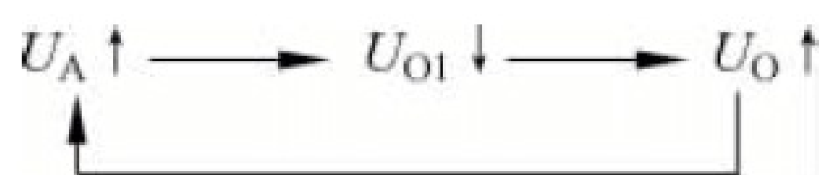


图 9.2.5 断开电位 U_{T-}

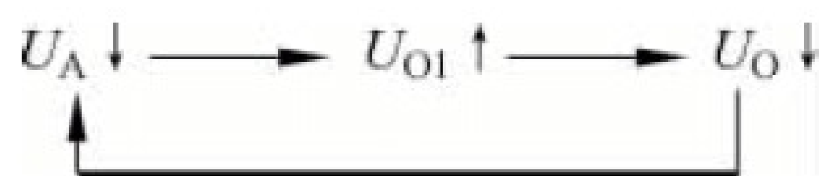
显然，为保证 U_{T-} 为正值，有 $R_2>R_1$，可求出回差电压为

$$\Delta U_T=U_{T+}-U_{T-}=\left(1+\frac{R_1}{R_2}\right)\frac{1}{2}U_{DD}-\left(1-\frac{R_1}{R_2}\right)\frac{1}{2}U_{DD}=\frac{R_1}{R_2}U_{DD} \tag{9.2.4}$$

当然，脉冲处理电路的最主要的目标是获得高性能的脉冲。在如图 9.2.3 所示电路中，U_I从低电平上升，当 $U_I=U_{T+}$，$U_A=1/2U_{DD}$时，为确保电路迅速进入额定状态，将引发一个正反馈过程，具体如下：



类似地，在如图 9.2.3 所示电路中，U_I从高电平往下降，当 $U_I=U_{T-}$，$U_A=1/2U_{DD}$时，为确保电路迅速进入额定状态，将引发另一个正反馈过程，具体如下：



上面两个正反馈过程确保了如图 9.2.3 所示脉冲电路输出脉冲的上升沿、下降沿均非常陡峭。

【例 9.2.1】 在如图 9.2.3 所示电路中，如果要求接通电位 $U_{T+}=7\text{V}$，回差电压 $\Delta U_T=4\text{V}$，求 R_1、R_2及 U_{DD}的值。

解 由式(9.2.2)和式(9.2.4)，有

$$\begin{cases} U_{T+}=\left(1+\dfrac{R_1}{R_2}\right)\dfrac{1}{2}U_{DD}=7\text{V} \\ \Delta U_T=\dfrac{R_1}{R_2}U_{DD}=4\text{V} \end{cases} \tag{9.2.5}$$

求解上面的方程组，有

$$R_1/R_2 = 0.4, \quad U_{DD} = 10V$$

根据集成门电路应用要求，反相器 G2 输出高电平时的负载电流不得超过最大允许值 $|I_{OHMAX}|$，假定使用 CC4069 集成六反相器 CMOS 门，查手册，当 $U_{DD}=10V$ 时，$|I_{OHMAX}|=1.3mA$。

由图 9.2.5，有

$$R_2 > \frac{U_O - U_A}{i} = \frac{10-5}{1.3} \approx 3.85k\Omega$$

可选 $R_1=30k\Omega$，$R_2=75k\Omega$。

当然，也可用 TTL 门电路组成施密特触发器，参考电路如图 9.2.6 所示。有兴趣的读者可自己分析其工作原理。

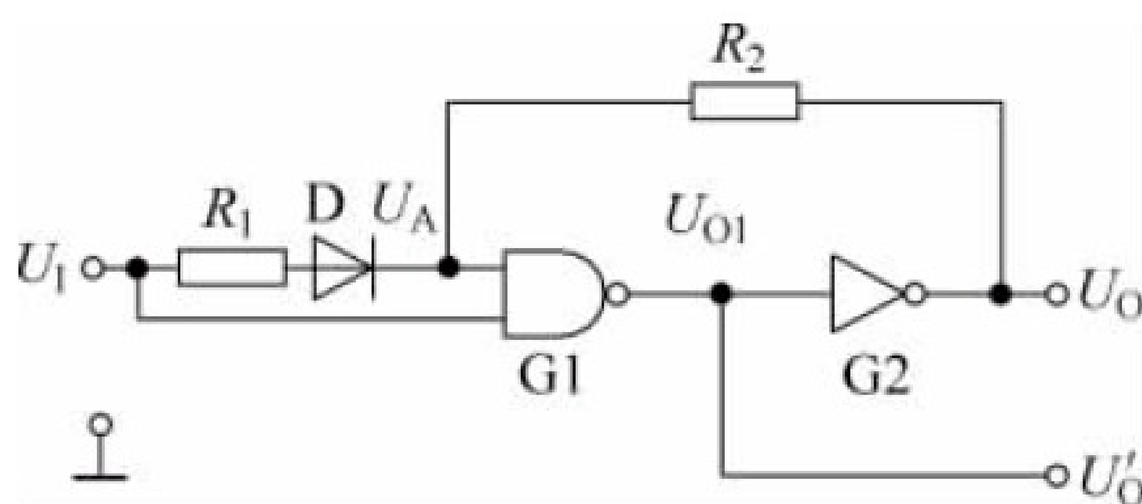


图 9.2.6　用 TTL 门电路组成的施密特触发器

9.2.3　集成施密特触发器

为提高集成度，改善集成电路性能，集成的施密特触发器当然不会采用如图 9.2.3 和图 9.2.7 所示的用门电路构成施密特触发器的方法构成施密特触发器。集成的施密特触发器一般直接采用三极管、MOS 管等器件直接实现施密特触发器，通过附加输入、输出等接口电路提高集成电路的接口性能。

如图 9.2.7 所示为用晶体管构成的施密特触发器电路。当输入 $U_I=0$ 时，T_1 截止，T_1 集电极为高电平，T_2 饱和，$U_O=0$。当输入 $U_I=1$ 时，T_1 饱和，T_1 集电极为低电平，T_2 截止，$U_O=1$。

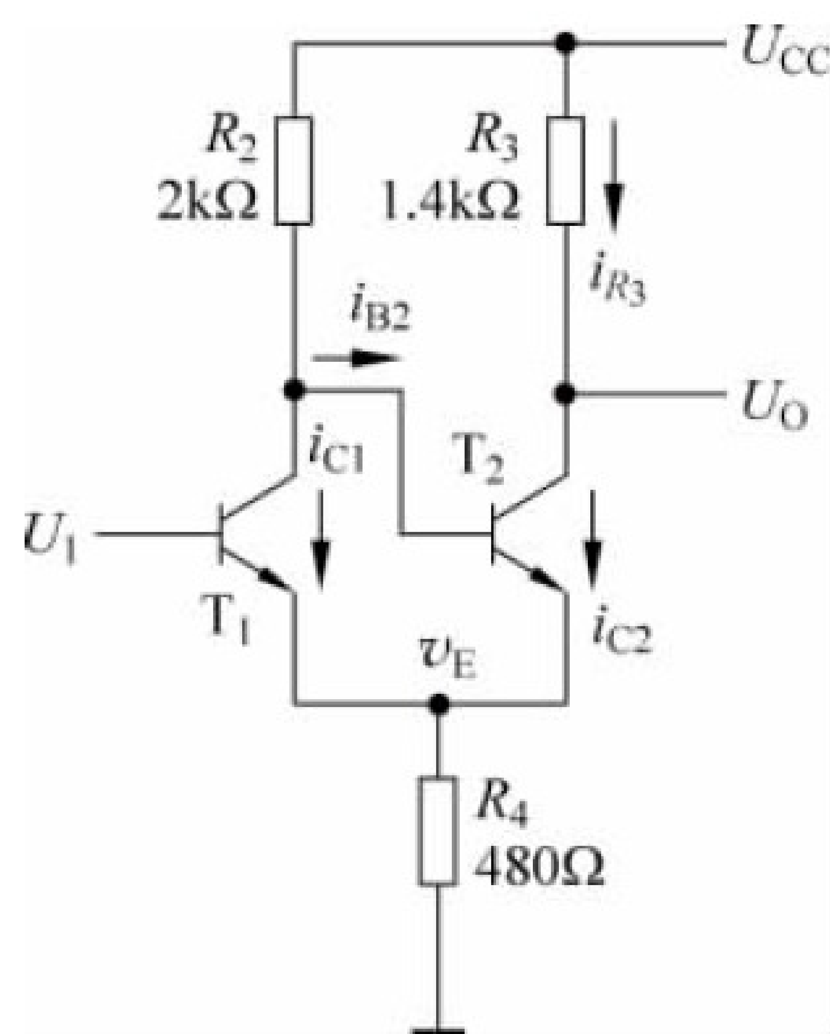


图 9.2.7　施密特触发器

当 U_I 从低电平开始上升时，T_1 保持截止，T_2 导通，U_O 将暂时保持 0 不变。当 U_I 上升到 U_{T+} 时，T_1 导通，T_2 截止，$U_O=1$。有

$$U_{T+} \approx i_{C2} \times R_4 + U_{BEON(T1)}$$

当 U_I 从高电平开始下降时，T_1 保持导通，T_2 截止，U_O 将暂时保持 1 不变。当 U_I 从高电平下降到 U_{T-} 时，T_1 截止，T_2 导通，$U_O=0$。有

$$U_{T-} \approx i_{C1} \times R_4 + U_{BEON(T1)}$$

电阻 $R_2 > R_3$，因此，T_2管的饱和电流 i_{C2}大于 T_1管的饱和电流 i_{C1}，满足 $U_{T+} > U_{T-}$。

可见，如图 9.2.7 所示用晶体管构成的电路具有施密特触发器的特性，为施密特触发器电路。

如图 9.2.8 所示为集成施密特触发器 74LS13 的原理图。图中包含了如图 9.2.7 所示的施密特触发器电路。此外，该电路在输入级增加了一个由二极管构成的 4 输入与门电路；在输出级增加了 TTL 电路的标准输出级电路，该输出电路具有反相器的逻辑功能，因此，如图 9.2.8 所示电路为施密特触发的与非门电路。

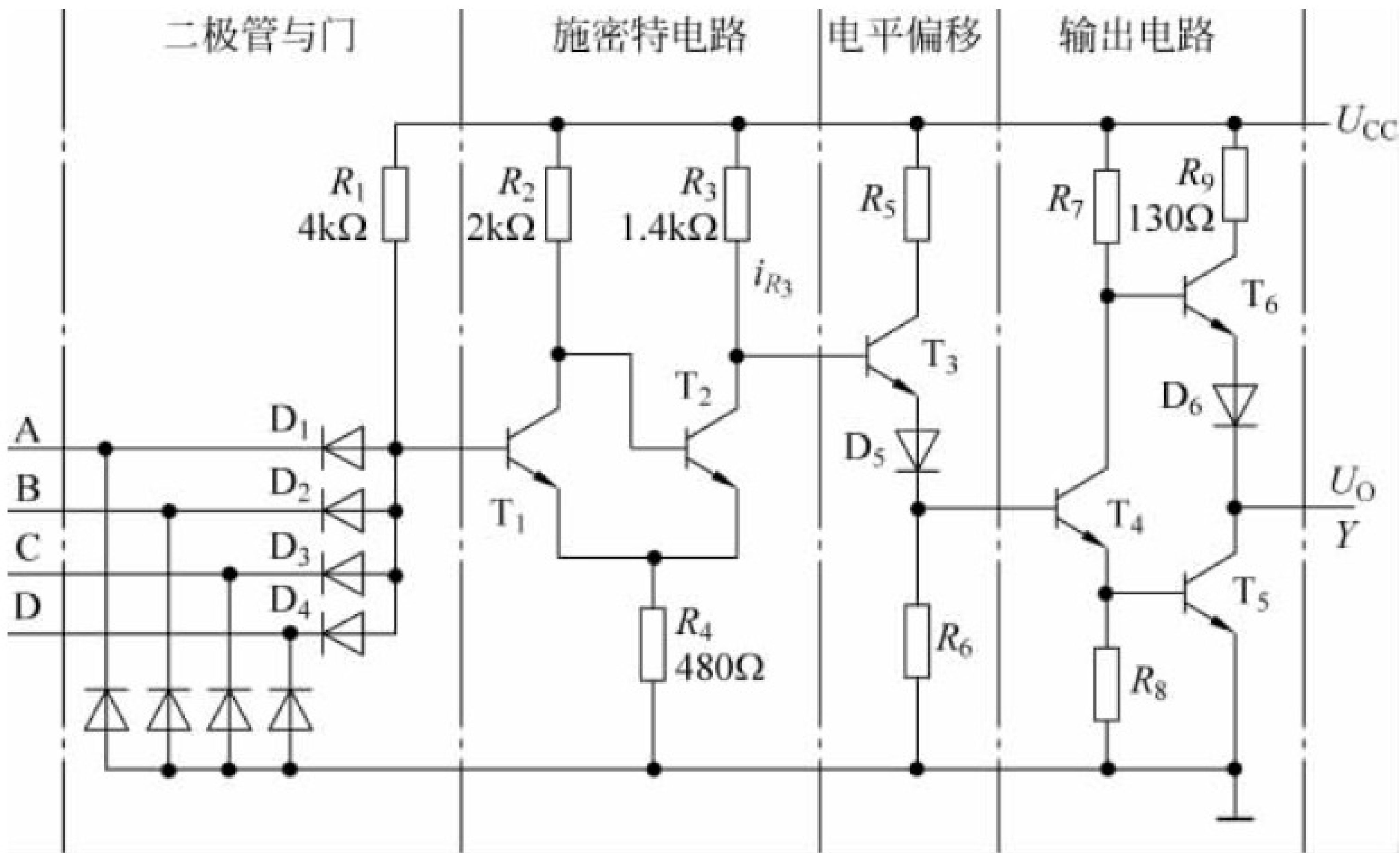


图 9.2.8 集成施密特触发器

施密特触发器应用十分广泛，集成的施密特触发器包括 TTL、CMOS 两大类。TTL 集成施密特触发器的产品主要有 74LS14（集成六施密特反相器）、74LS132、74LS13 等。国产 CMOS 集成施密特触发器的产品主要有 CC40106（集成六施密特反相器）、CC4093 等。关于 CMOS 集成施密特触发器的结构，有兴趣的读者可参考相关书籍。

9.2.4 施密特触发器的应用

施密特触发器的主要应用有：

1. 用作整形

如图 9.2.9 所示为施密特触发器用作整形的应用。它可以将不规则的信号变换为或者说整形为矩形脉冲。

如图 9.2.9 所示输入信号 U_I的波形，当 $0 \leqslant t < t_1$时，U_I虽然在增长，但小于接通电位 U_{T+}，由电压传输特性可知，触发器输出 $U_O = 0$；当 $t_1 \leqslant t < t_3$时，U_I先增长，$U_I > U_{T+}$，由电压传输特性可知，触发器输出 $U_O = 1$；U_I增长到最大值以后，开始下降，但依然有 $U_I > U_{T-}$，所

以，触发器保持输出 $U_O=1$ 不变；当 $t\geqslant t_3$ 时，有 $U_I<U_{T-}$，触发器输出 $U_O=0$，所以，可得出触发器输出如图示，为较理想的矩形波。

2. 用于脉冲鉴幅

从图 9.2.10 中看出，若将一系列幅度各异的脉冲信号加到施密特触发器的输入端时，只有那些幅度大于 U_{T+} 的脉冲才会在输出端产生输出信号。所以，施密特触发器能将幅度大于 U_{T+} 的脉冲输出，即具有脉冲鉴幅的能力。

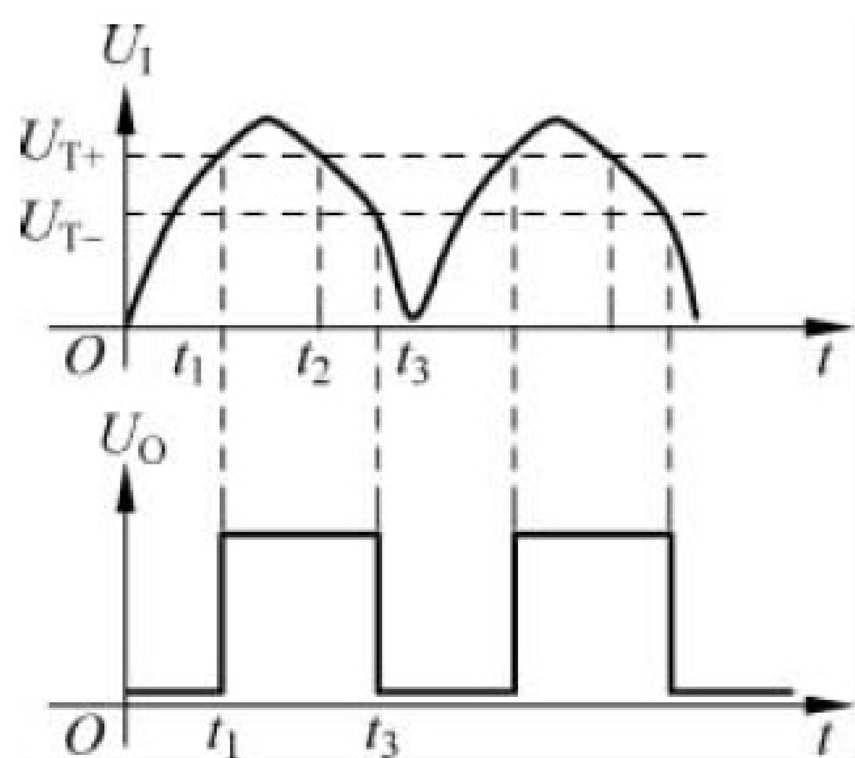


图 9.2.9　施密特触发器的波形

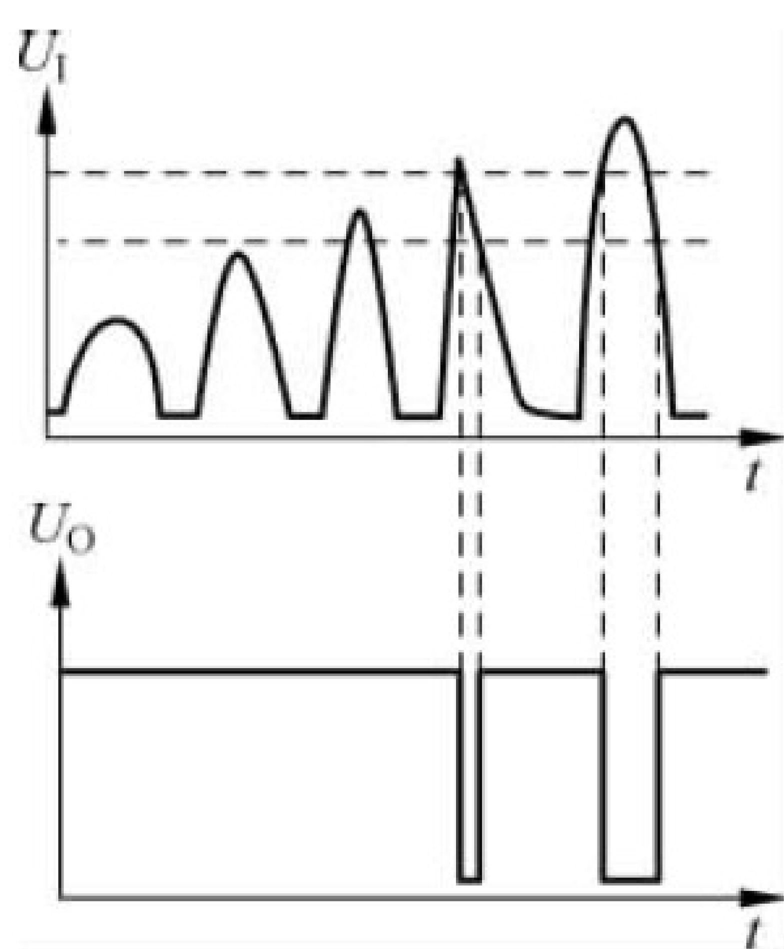


图 9.2.10　施密特触发器用于脉冲鉴幅

用施密特触发器还可组成脉冲展宽器、多谐振荡器等。如何用施密特触发器组成多谐振荡器将在后面介绍。

复习与思考

9.2.1　能否用施密特触发器存储 1 位二值代码？

9.2.2　如图 9.2.7 所示施密特触发器中电阻 R_2 可否小于 R_3？

9.2.3　如图 9.2.8 所示的施密特触发的与非门电路能否当作普通的 TTL 与非门使用？

9.2.4　如图 9.2.7 所示电路输出状态切换时是否存在正反馈？若存在，请画出反馈过程的示意图。

9.3　单稳态触发器



单稳态触发器的特点是：它具有一个稳态、一个暂稳态，而且在无触发脉冲作用时，电路处于稳态。当触发器脉冲触发时，电路能够从稳态翻转到暂稳态，在暂稳态维持一段时间以后，电路能够返回稳态。

单稳态触发器的暂稳态维持时间的长短只取决于电路本身的参数，而与触发脉冲的幅度和宽度无关。

9.3.1 用门电路组成的单稳态触发器

单稳态触发器的暂稳态的“暂稳”特点一般是通过 RC 电路的充、放电来实现的。依照 RC 电路的不同接法，可将单稳态触发器分为微分型单稳态触发器和积分型单稳态触发器两种。

如图 9.3.1 所示为用 CMOS 门电路构成的微分型单稳态触发器。

稳态时，电容相当于开路，电路只具有一个稳态 $U_O=0$，$U_{O1}=1$。对 CMOS 门电路，高电平为电源值 U_{DD}，阈值电平约为 $1/2U_{DD}$，输入电流为 0，因此，稳态下，电容 C 上没有电荷。$U_2=U_{DD}$。

假定在 $t=0$ 时刻(电路已达稳态)在触发器的 U_I 加上如图 9.3.2(a)所示的正脉冲，则在 $t<t_1$ 时，电路依然处于稳态；当 $t=t_{1+}$ 时，U_I 瞬间跳变到 1，由于电容两端的电压不能跳变，所以，U_1 瞬间也跳变到 1，使 $U_{O1}=0$；$U_{O1}=0$ 使 U_2 瞬间也跳变到 0，$U_O=1$，电路进入暂态。

$U_O=1$，反馈到或门 G1 的输入端，维持 $U_{O1}=0$ 不变。电路的暂态不会随着触发信号的撤除而改变。之后，正脉冲撤除，电路维持 $U_O=1$ 不变。

电路进入暂态以后，电源 U_{DD} 将对电容 C 进行充电，U_2 的电位开始上升，当 U_2 的电位上升到非门阈值电压 $1/2U_{DD}$ 时，$U_O=0$，电路重新回到稳态，因此，其输出波形如图 9.3.2(b)所示。

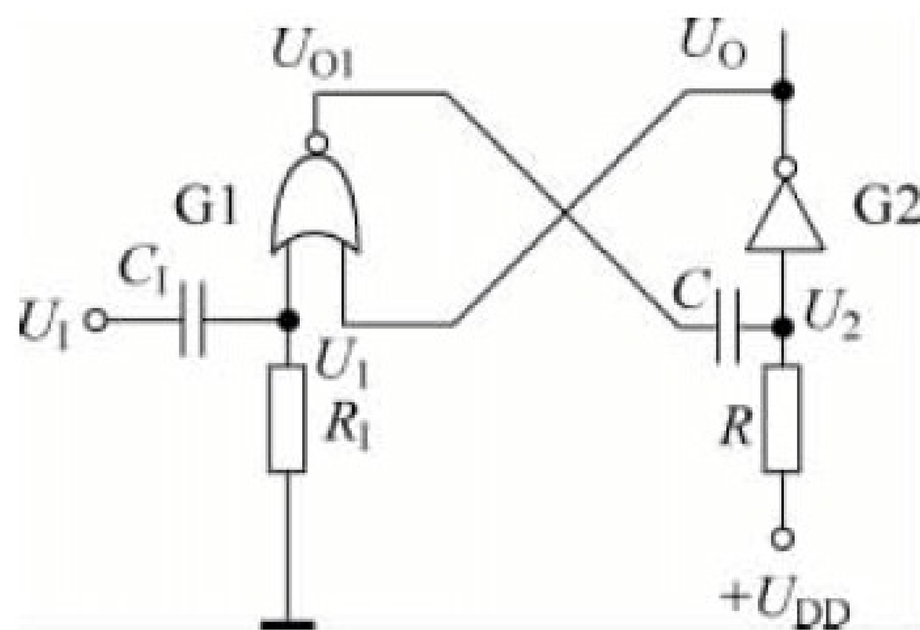


图 9.3.1 微分型单稳态触发器

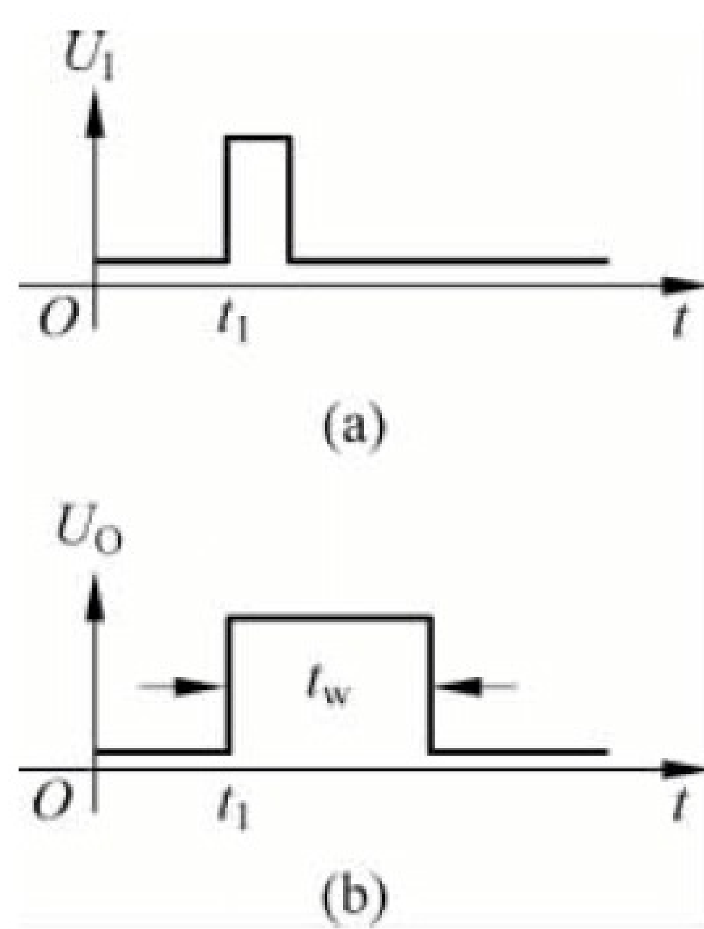


图 9.3.2 单稳触发器波形

如图 9.3.1 所示的单稳态触发器的暂稳态维持时间取决于 RC 微分电路，有

$$t_w \approx 0.69RC \tag{9.3.1}$$

当然，可通过一阶 RC 电路三要素法直接求出 t_w。一阶 RC 电路电容 C 上的响应

$$U_2=U_S+(U_0-U_S)e^{-\frac{t}{\tau}} \tag{9.3.2}$$

式中，U_S 为稳态值；U_0 为初始值；τ 为一阶 RC 电路的时间常数。

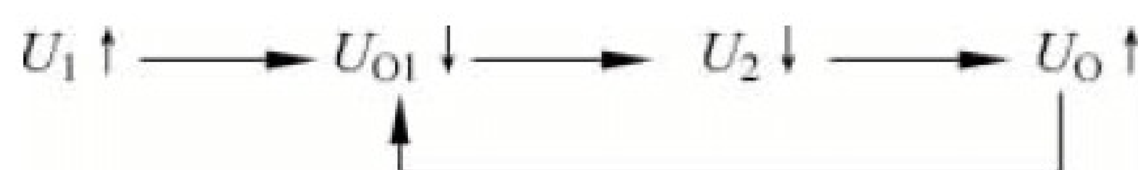
如图 9.3.1 所示一阶 RC 电路中，$\tau=RC$，$U_S=U_{DD}$，$U_0=0$。当 $U_2=1/2U_{DD}$ 时，单稳态触发器输出离开暂稳态，进入稳态。

将上面参数代入式(9.3.2)求解,有

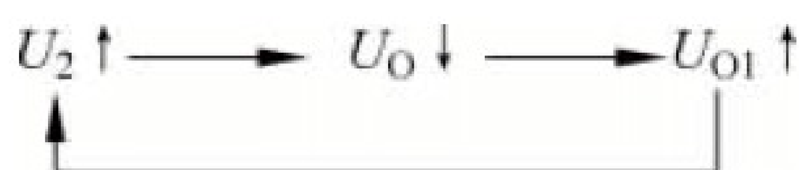
$$t_w = \tau \ln \frac{U_S - U_0}{U_S - U_2} = RC\ln 2 \approx 0.69RC$$

当然,稳态时电容 C 上是没有电荷的。单稳态触发器输出离开暂稳态进入稳态后,内部电路并没有稳定,需要等到电容 C 放电完毕后电路才恢复为起始的稳态。一般认为经过3～5 倍时间常数的时间后,RC 电路基本达到稳态。

单稳态触发器为常用的脉冲处理电路之一,其最主要的目标之一也是获得高性能的脉冲。如图 9.3.1 所示电路中当 U_I正脉冲到来时,为确保电路迅速进入暂态,将引发一个正反馈过程,具体如下:



类似地,如图 9.3.1 所示电路进入暂态后,电容 C 将被充电。当 U_2的电位上升到非门阈值电压 $1/2U_{DD}$时,单稳态触发器输出将离开暂稳态,进入稳态。为确保电路迅速进入稳态,将引发另一个正反馈过程,具体如下:



上面两个正反馈过程确保了如图 9.3.1 所示电路输出脉冲的上升沿、下降沿均非常陡峭。

如图 9.3.1 所示电路的特点为窄正脉冲触发,稳态时电容 C 上没有电荷。当使用宽正脉冲触发时,触发脉冲宽度大于暂态脉冲宽度,电路依然可以工作,但破坏了暂态切换到稳态时的正反馈过程,输出脉冲下降沿较差,故建议不使用宽正脉冲触发。

如图 9.3.3 所示为用 TTL 门电路构成的单稳态触发器,输出为负脉冲。该电路的特点为宽正脉冲触发,要求触发脉冲宽度大于暂态脉冲宽度。有兴趣的读者请自己分析其工作原理。该电路的暂稳态主要通过 RC 积分电路实现,因此,常把该电路称为积分型单稳态触发器。

显然,如图 9.3.3 所示积分型单稳态触发器所有的输出(含中间输出)均没有反馈到输入端,没有反馈支路,当然也就不存在提高脉冲性能的正反馈过程,输出波形的边沿比较差,在脉冲处理应用中不如积分型单稳态触发器应用广泛。

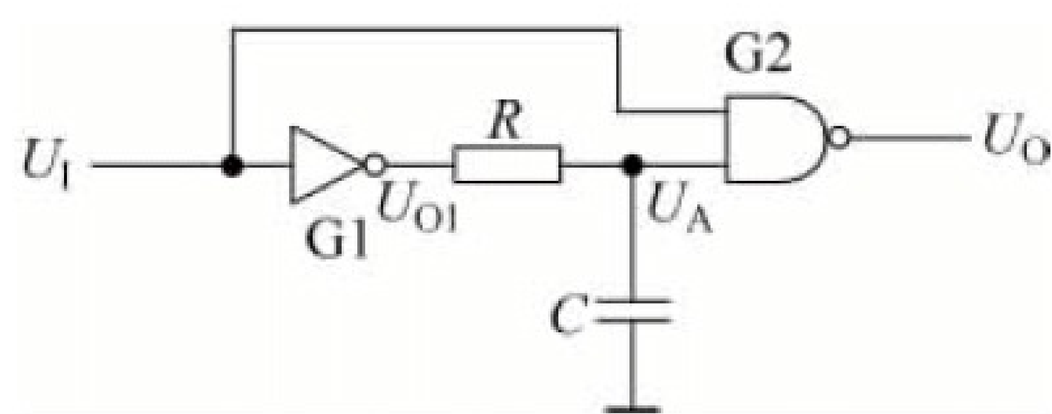


图 9.3.3 积分型单稳态触发器

当然,如图 9.3.3 所示的积分型单稳态触发器采用宽正脉冲触发,而数字系统中的干扰信号一般均为窄脉冲,因此,如图 9.3.3 所示的积分型单稳态触发器具有更好的抗干扰性能。

如图 9.3.4 所示电路为改进积分型单稳态触发器。和如图 9.3.3 所示电路相比,该电路的最大改进特点为窄负脉冲触发。触发器进入暂态后输出为 0,反馈到与非门 G3 的输入端,确保窄负脉冲撤销后与非门 G3 输出为 1,维持暂态不变。可见,如图 9.3.4 所示电

路中，输出反馈到了输入端，保证了窄负脉冲触发。有兴趣的读者可进一步分析其工作原理。

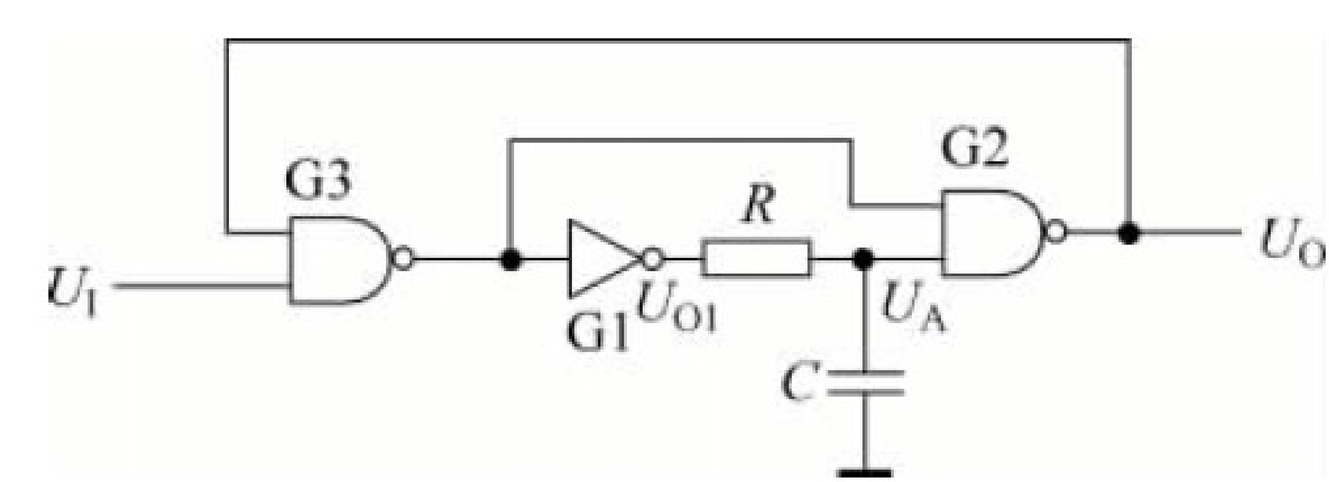


图 9.3.4　改进积分型单稳态触发器

9.3.2　集成单稳态触发器

单稳态触发器广泛地应用于脉冲的整形、延时和定时等，相应的集成芯片较多，CMOS 系列有 CC14528，TTL 系列有 74LS121、74LS122、74LS123 等。

目前使用的集成单稳态触发器有不可重复触发型和可重复触发型两大类。不可重复触发型单稳态触发器一旦被触发进入暂态后，再在输入端加触发脉冲不会影响电路的工作过程，必须在本次暂稳态过程结束后才可接受下一个触发脉冲。可重复触发型单稳态触发器一旦被触发进入暂态后，稳态过程未结束前再在输入端加触发脉冲，电路将重新被触发，单稳态触发器重新开始暂稳态过程。74LS121 为不可重复触发的单稳态触发器，74LS122、74LS123 为可重复触发的单稳态触发器。

表 9.3.1　74LS121 功能表

输　入			输　出	
A_1	A_2	B	U_O	U'_O
0	×	1	0	1
×	0	1	0	1
×	×	0	0	1
1	1	×	0	1
1	↓	1	⊓	⊔
↓	1	1	⊓	⊔
↓	↓	1	⊓	⊔
0	×	↑	⊓	⊔
×	0	↑	⊓	⊔

如图 9.3.5 所示为集成不可重复触发单稳态触发器 74LS121 的引脚图。图中，A_2、A_1 为两个负脉冲触发输入端，B 为正脉冲触发输入端，U_O 为单稳态触发器输出端，U'_O为反相

输出端，R、C 为外部电阻、电容接入端，R_{in} 为内部电阻连接端。未列出的序号引脚为空引脚。功能表如表 9.3.1 所示。从功能表前 4 行可看出，74LS121 稳态下输出为 0。功能表第 5～7 行描述了负脉冲触发时的输入状态，第 8～9 行描述了正脉冲触发时的输入状态。

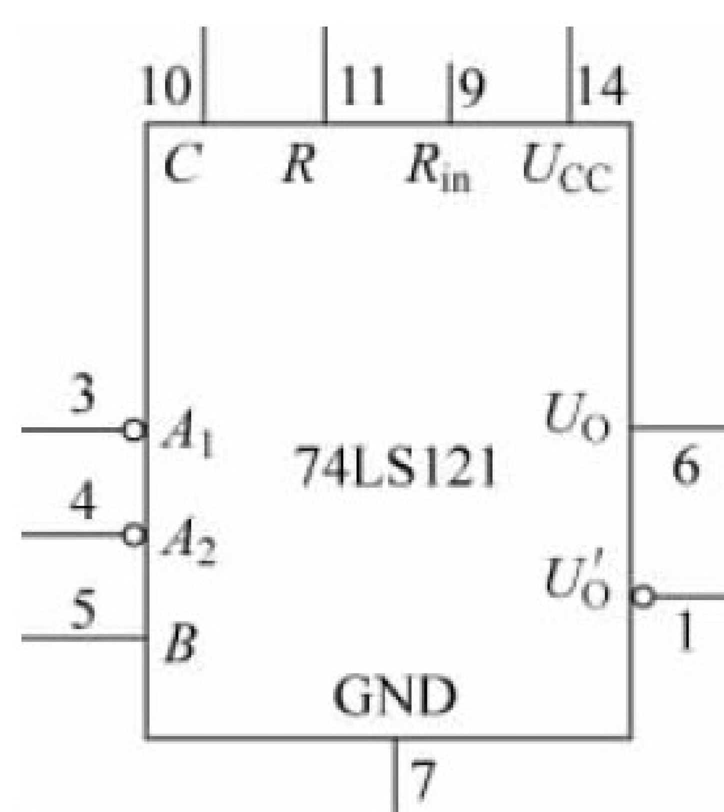


图 9.3.5　74LS121 引脚图

74LS121 简化原理图如图 9.3.6 所示。虚框(2)中的电路块和外接的电阻及电容共同构成了微分型的单稳态触发器。该单稳态触发器的电路结构和如图 9.3.1 所示的微分型单稳态触发器基本相同，稳态下输出为 0，窄正脉冲触发。当引脚 R、C 间接入外部电容 C，引脚 R 经外部电阻 R 接电源时，暂态持续时间 $t_w \approx 0.69RC$。

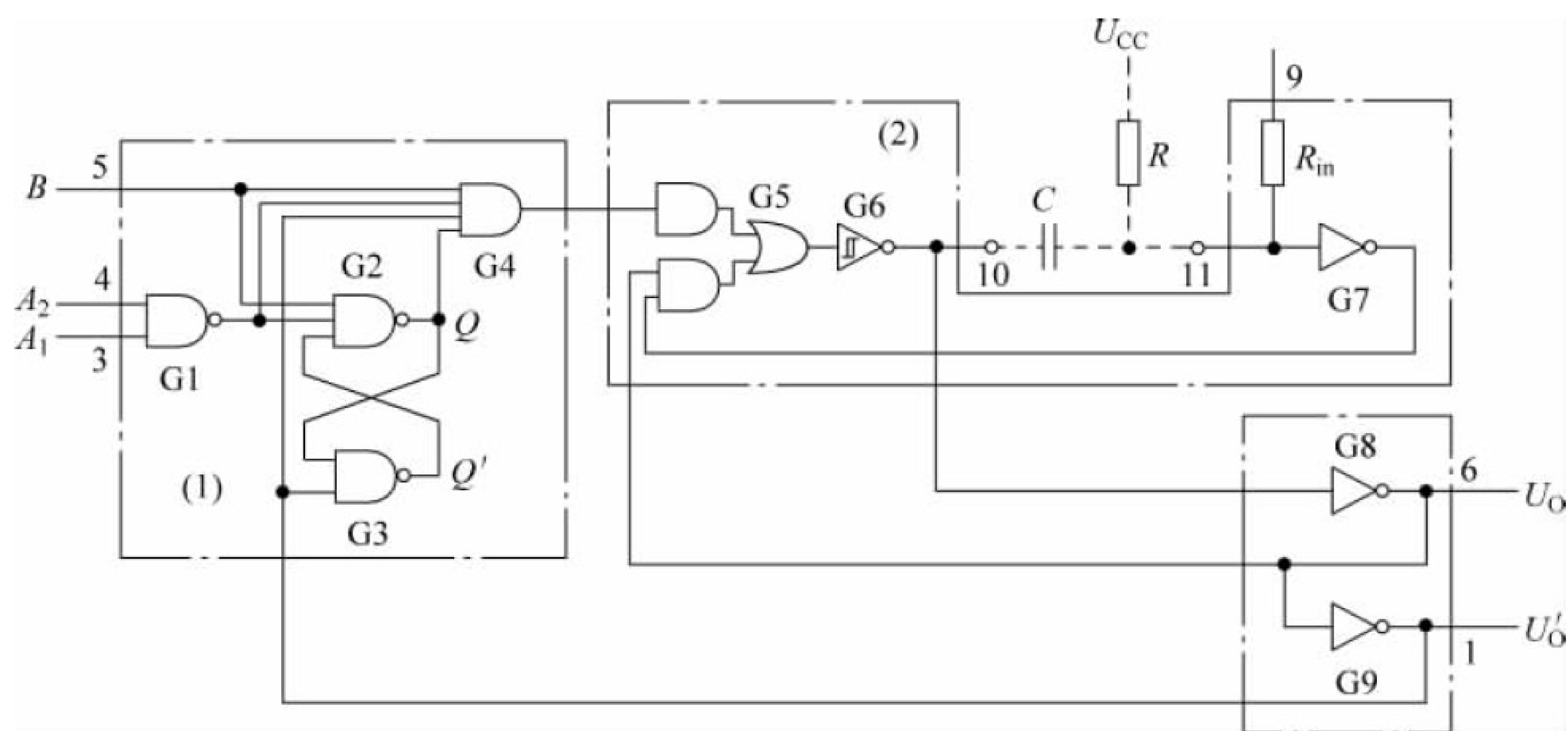


图 9.3.6　74LS121 简化原理图

图 9.3.6 中，虚框(1)中的电路块实现了正、负脉冲两种触发方式。B 为正脉冲触发输入端，直接经与门接微分型单稳态触发器的触发输入端。有兴趣的读者可进一步分析负脉冲触发的实现方法。

利用 74LS121 构成的窄负脉冲触发的单稳态触发器的典型电路如图 9.3.7 所示。图中，A_1 接入窄负脉冲，A_2 及 B 接 1。引脚 R、C 间接入外部电容 C，引脚 R 经外部电阻 R 接电源。U_O 输出正脉冲，$t_w \approx 0.69RC$。

通常，R 的取值为 2～30kΩ。C 的取值为 10pF～10μF，由此得到的 t_w 范围为 20ns～200ms。

利用自带电阻实现的窄正脉冲触发的单稳态触发器的典型电路如图 9.3.8 所示。图中，A_2、A_1 接地，B 接入窄正脉冲。引脚 R、C 间接入外部电容 C，引脚 R_{in} 接电源，使用内部电阻 R_{in}。当然，内部电阻 R_{in} 阻值不大，约为 2kΩ，因此，当希望获得较宽输出脉冲时，仍需外接电阻。

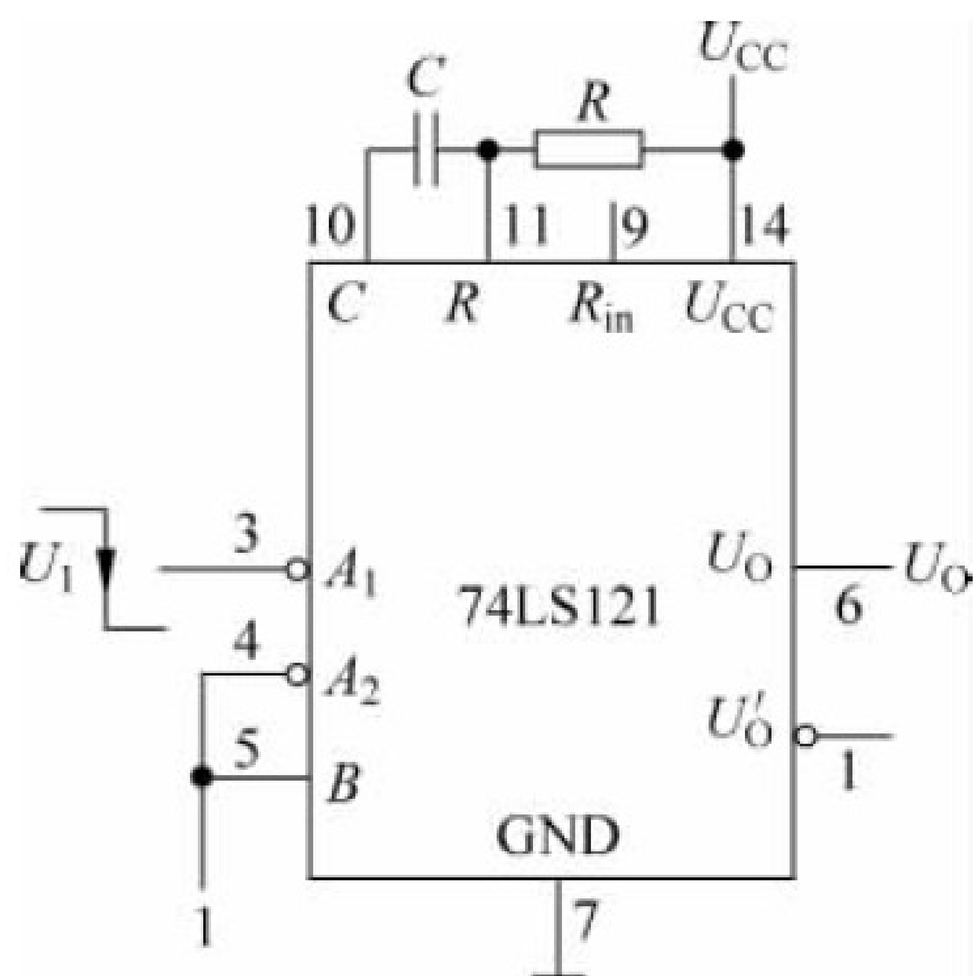


图 9.3.7　典型应用电路 1

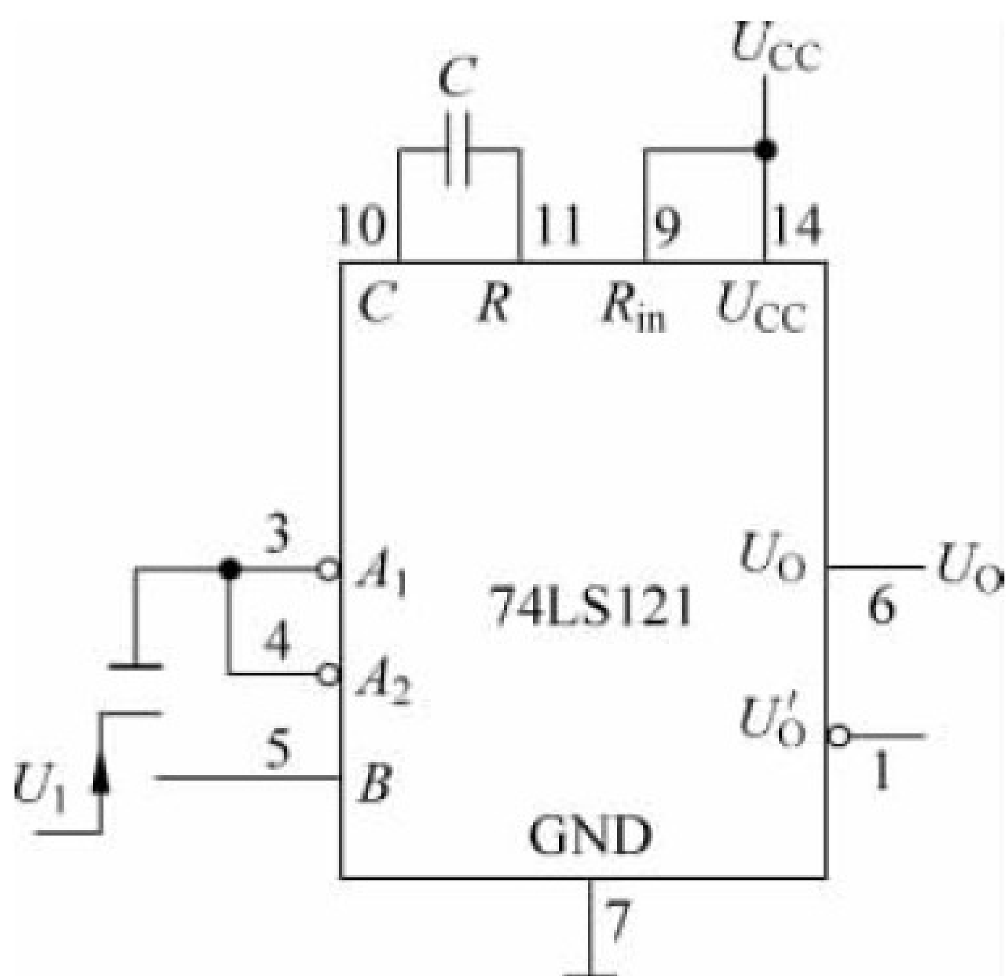


图 9.3.8　典型应用电路 2

可总结 74LS121 的典型应用方法如表 9.3.2 所示。

表 9.3.2　74LS121 的典型应用方法

触发输入	参数设置	U_O
A_2、A_1的任意一个接入窄负脉冲，另一个及 B 接 1（或 A_2、A_1接地，B 接入窄正脉冲）	R、C 间接外部电容 C，使用外部电阻 R	正脉冲 $t_w \approx 0.69RC$
	R、C 间接外部电容 C，使用内部电阻 R_{in}	正脉冲 $t_w \approx 0.69R_{in}C$

复习与思考

9.3.1　试总结如图 9.3.1 所示微分型单稳态触发器及如图 9.3.3 所示积分型单稳态触发器各自的优缺点。

9.3.2　如图 9.3.3 所示积分型单稳态触发器有无正反馈过程？为什么？

9.4　多谐振荡器



多谐振荡器是一种能够产生一定频率和一定宽度的矩形波的电路。它不需要外加输入信号的作用，它没有稳态，所以又称为无稳态电路。多谐振荡器可产生时钟信号，是绝大多数数字系统的必备电路。

9.4.1　用门电路构成多谐振荡器

构成多谐振荡器的方法较多，如图 9.4.1 所示为用 TTL 门电路组成的对称式多谐振荡器电路，由两个含反馈支路的反相器 G1、G2 经耦合电容 C_1、C_2连接而成。

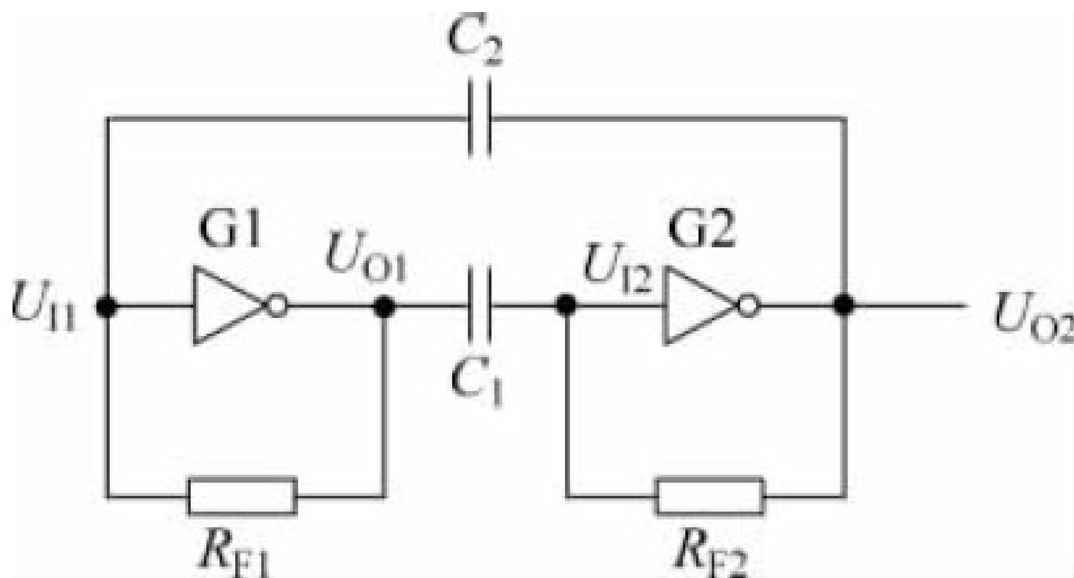


图 9.4.1　对称式多谐振荡器

合理选择电路参数，使两个反相器静态下工作在

转折区(阈值电平附近),确保上电微弱变化可使反相器状态变化。

该电路为无稳态电路,输出 U_{O2}将在"0""1"两个暂态间来回切换。当然,多谐振荡器中的暂态具有一定的稳态特性,为暂稳态。下面具体地分析对称式多谐振荡器的暂态过程。

初始上电反相器工作在转折区,一个微弱的正变化将使 $U_{I1}=1$。由于 U_{I1}为高电平,电路进入第 1 个暂稳态,有

$$U_{I1}=1\rightarrow U_{O1}=0\rightarrow U_{I2}=0\rightarrow U_{O2}=1$$

当然,这个微弱的正变化只是干扰,如果振荡器不能迅速脱离转折区,将可能会有一个微弱的负变化使 $U_{I1}=0$,从而使电路进入另一个暂态。事实上,当 $U_{I1}\uparrow$ 并达到阈值电平时,将引发一个正反馈过程,使电路快速进入第 1 个暂稳态。正反馈过程具体如下:

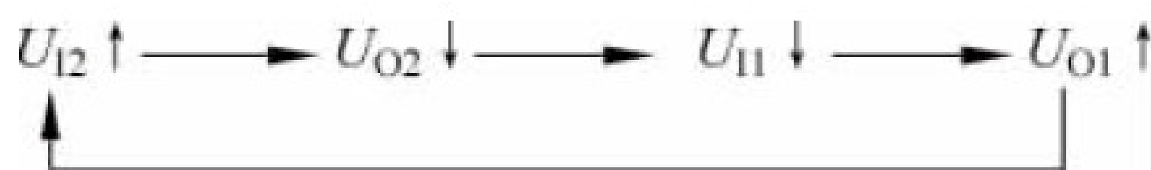


上面的正反馈过程确保了 U_{I1}迅速脱离转折区,快速进入第 1 个暂稳态,确保了输出脉冲 U_{O2}的上升沿非常陡峭。

此时,U_{I1}、U_{O2}均为高电平,U_{O1}、U_{I2}均为低电平。U_{O2}将通过 R_{F2}对电容 C_1充电,使 U_{I2}电位上升。电容 C_2将通过 R_{F1}放电,U_{I1}电位将下降。

电路结构上保证了电容 C_1的充电时间小于电容 C_2的放电时间。由于 C_1充电更快,U_{I2}电位将率先上升到 U_{TH},输出 U_{O2}翻转到 0,进入第 2 个暂稳态。

类似地,当 $U_{I2}\uparrow$ 并达到阈值电平时,将引发一个新的正反馈过程,使电路快速进入第 2 个暂稳态。正反馈过程具体如下:



上面的正反馈过程确保了 U_{I2}迅速脱离转折区,快速进入第 2 个暂稳态,确保了输出脉冲 U_{O2}的下降沿非常陡峭。

此时,U_{I1}、U_{O2}均为低电平,U_{O1}、U_{I2}均为高电平。U_{O1}将通过 R_{F1}对电容 C_2充电、电容 C_1将通过 R_{F2}放电,将使 U_{I2}电位下降,U_{I1}电位上升。

电路结构上保证了电容 C_2的充电时间小于电容 C_1的放电时间。由于 C_2充电更快,因此 U_{I1}电位将率先上升到 U_{TH},引发和上电微弱正干扰类似的正反馈,快速进入第 1 个暂稳态。

综上所述,如图 9.4.1 所示电路在两个暂稳态间来回切换,在输出端产生了固定频率和固定宽度的矩形波,为典型的多谐振荡器电路之一。

第 1 个暂稳态持续的时间为 U_{I2}电位从低电平开始上升到 U_{TH}时 C_1的电容充电时间,第 2 个暂稳态持续的时间为 U_{I1}电位从低电平开始上升到 U_{TH}时电容 C_2的充电时间。如图 9.4.1 所示电路为对称式多谐振荡器,电路结构对称,参数相同,两个暂稳态持续的时间相同。关于对称式多谐振荡器输出矩形波频率的具体计算方法,有兴趣的读者可参考相关书籍。

既然如图 9.4.1 所示电路结构对称,参数相同。读者可能迷惑,如图 9.4.1 所示电路结

构上既保证了电容 C_2 的充电时间小于电容 C_1 的放电时间，还保证了电容 C_1 的充电时间小于电容 C_2 的放电时间，怎么可能呢?

读者不要忘记 TTL 门电路的特点。当 TTL 门电路输入为高电平时，输入电流实际方向为流入门电路，为反向穿透电流，不会流向电容，也就不会对电容充电。当输入为低电平时，输入电流实际方向从门电路流出，将流向电容并对电容充电。

在如图 9.4.1 所示对称式多谐振荡器电路中，当对电容进行充电时，相关输入端均为低电平，输入电流从门电路流出并对电容充电。可见，如图 9.4.1 所示对称式多谐振荡器电路中，对电容进行充电时，有两条充电支路充电；放电时，只有一条放电支路，因此，保证了电容的充电时间快于电容的放电时间。

如图 9.4.1 所示对称式多谐振荡器电路可进一步简化。静态时反相器 G1 工作在转折区，去掉电容 C_1 后，G1 的输出接 G2 的输入，依旧可确保静态时反相器 G2 工作在转折区。此外，只要存在反馈电容 C_2，电路就不可能有稳定状态，只能在两个暂稳态之间来回切换。可见，如图 9.4.1 所示对称式多谐振荡器电路中去掉电容 C_1、电阻 R_{F2} 依旧可构成多谐振荡器，用 CMOS 门电路实现的非对称式多谐振荡器参考电路如图 9.4.2 所示，有兴趣的读者可自己分析该电路的工作原理。

还可利用门电路的传输延迟时间构成多谐振荡器。如图 9.4.3 所示为用 3 个非门电路构成的环形多谐振荡器，由 3 个反相器首尾相连组成。

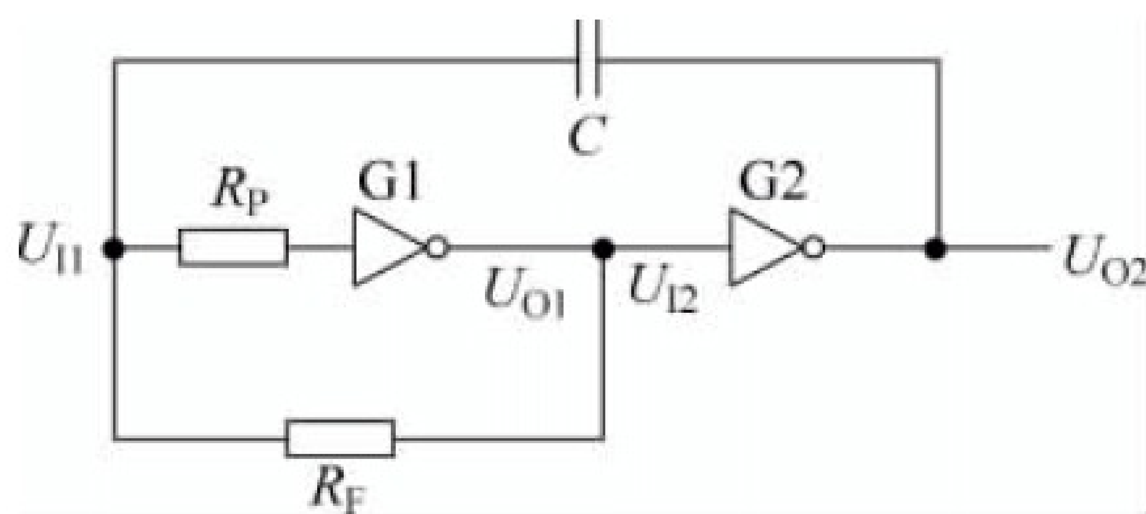


图 9.4.2　非对称式多谐振荡器

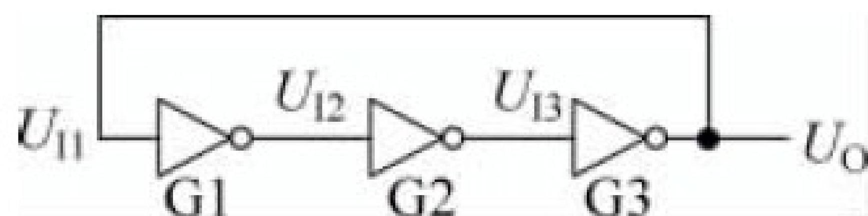


图 9.4.3　环形多谐振荡器

假设初始上电输出 U_O 为 0，则 $U_{I1}=U_O=0$，经 3 个门电路的传输时间，输出 U_O 将翻转到 1，之后，又经 3 个门电路的传输时间重新回到 0。可见，如图 9.4.3 所示环形多谐振荡器的输出为矩形波，周期为 $6t_{pd}$。

9.4.2　用施密特触发器构成多谐振荡器

可用施密特触发器组成多谐振荡器，参考电路如图 9.4.4 所示。

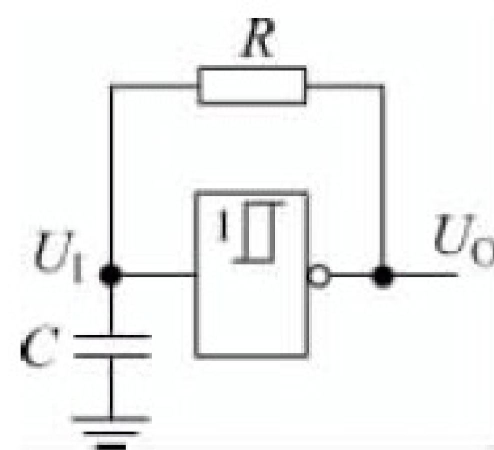


图 9.4.4　用施密特触发器组成的多谐振荡器

其工作原理分析如下：

初始上电，电容没有储存能量，电压 $U_I=0$，施密特反相器输出 $U_O=1$，为高电平。此时，$U_O=1$，$U_I=0$，U_O 通过电阻 R 对电容 C 充电，U_I 电压开始上升；当 U_I 上升到 U_{T+} 时，施密

特反相器翻转，输出 $U_O=0$，为低电平。此时，$U_O=0$，$U_I=U_{T+}$，电容 C 将放电，电压 U_I 开始下降；当 U_I 下降到 U_{T-} 时，施密特反相器翻转，输出 $U_O=1$；周而复始，可产生一定频率和一定宽度的矩形波，电压波形如图 9.4.5 所示。

对 CMOS 施密特反相器，有 $U_{OH}=U_{DD}$，$U_{OL}=0$，可算出多谐振荡器的振荡周期的计算公式为

$$\begin{aligned} T &= RC\ln\frac{U_{DD}-U_{T-}}{U_{DD}-U_{T+}}+RC\ln\frac{U_{T+}}{U_{T-}} \\ &= RC\ln\left(\frac{U_{DD}-U_{T-}}{U_{DD}-U_{T+}}\cdot\frac{U_{T+}}{U_{T-}}\right) \end{aligned} \tag{9.4.1}$$

可通过调节 R、C 的大小，改变振荡频率。

还可利用施密特触发器构成占空比可调的多谐振荡器，参考电路如图 9.4.6 所示。改变 R_2、R_1 的值即可改变占空比。

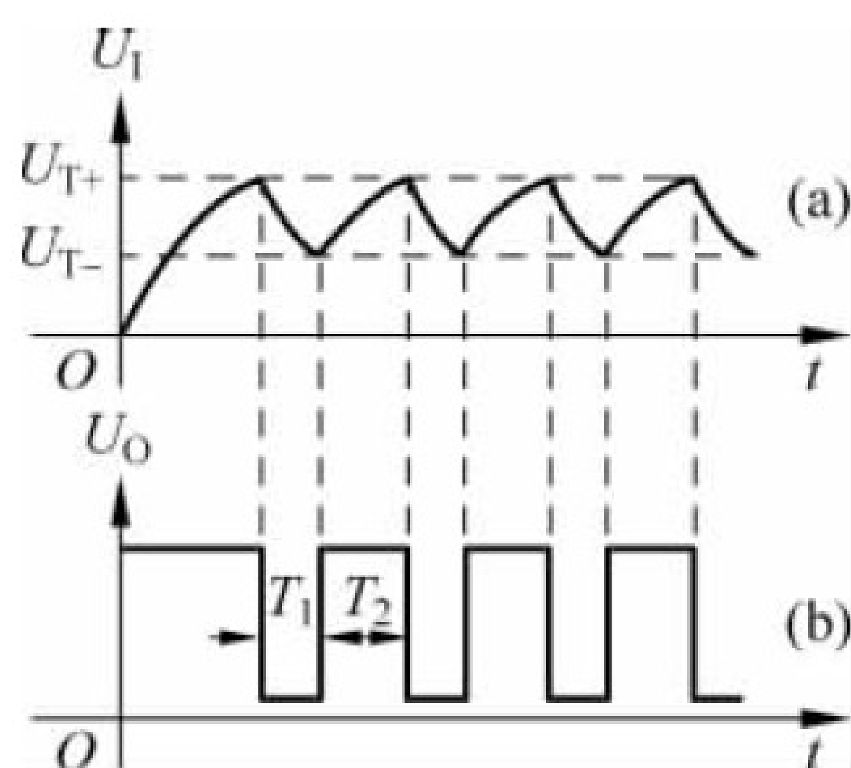


图 9.4.5　电压波形

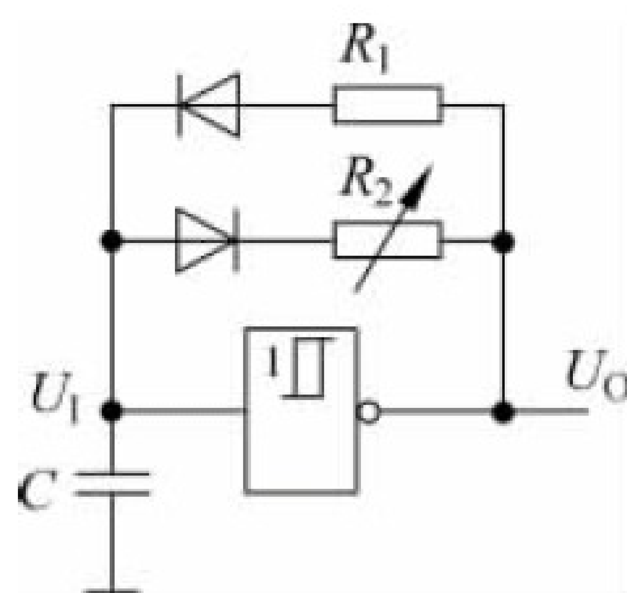


图 9.4.6　占空比可调的多谐振荡器

9.4.3　石英晶体多谐振荡器



在许多应用场合下，对多谐振荡器的振荡频率的稳定性有严格要求，而如图 9.4.1 和图 9.4.4 所示多谐振荡器的电容充、放电过程极易受外部干扰，门电路的阈值电压 U_{TH} 也不稳定，极易受电源电压和外部干扰的影响，因此，如图 9.4.1 和图 9.4.4 所示多谐振荡器难以满足较高频率稳定性的要求。

目前普遍采用的稳频方法是在多谐振荡器电路中接入石英晶体，组成石英晶体多谐振荡器，石英晶体的逻辑符号如图 9.4.7 所示。

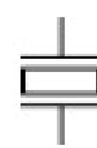


图 9.4.7　石英晶体的逻辑符号

石英晶体是利用具有压电效应的石英晶体片制成的。这种石英晶体薄片受到外加交变电场的作用时会产生机械振动，当交变电场的频率与石英晶体的固有频率相同时，振动便变得很强烈。利用这种特性，可以用石英谐振器取代谐振回路、滤波器等。由于石英谐振器具有体积小、重量轻、可靠性高、频率稳定度高等优点，被应用于家用电器和通信设备中。

石英晶体多谐振荡器的振荡频率取决于石英晶体的固有谐振频率，与外接电阻、电容

等元件无关。石英晶体的固有谐振频率由石英晶体的结晶方向和外形尺寸所决定，具有极高的频率稳定性，常把石英晶体称为石英晶体振荡器或石英谐振器，简称晶振。

利用石英晶体设计实现多谐振荡器的方法：选择固有谐振频率合适的石英晶体，将石英晶体接入如图 9.4.1 和图 9.4.2 所示的多谐振荡器电路中的反馈支路即可。用石英晶体实现的多谐振荡器参考电路如图 9.4.8 所示。

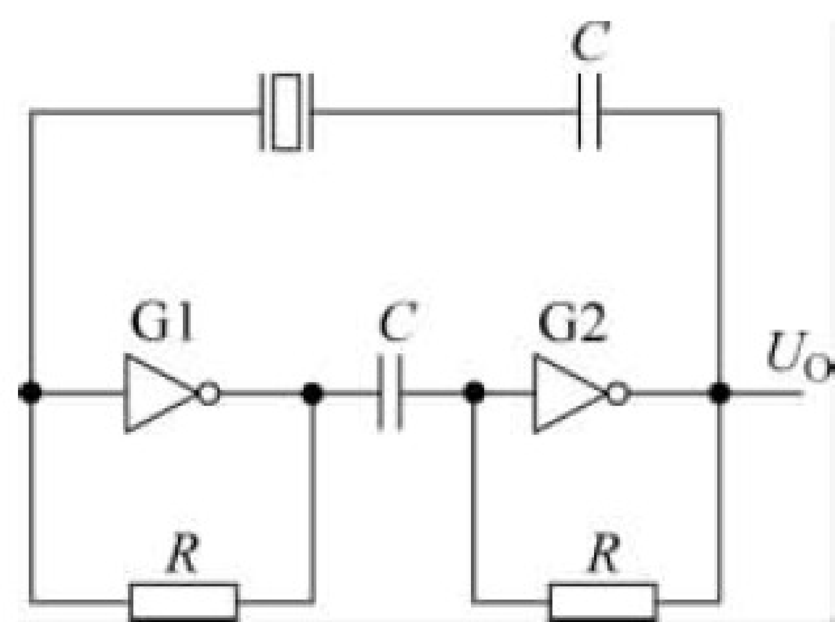


图 9.4.8 石英晶体多谐振荡器

复习与思考

9.4.1 如图 9.4.1 所示电路结构上对称，C_2的充电时间和放电时间是否相同？为什么？

9.4.2 为什么石英晶体能稳定振荡电路的振荡频率？

9.5 555 定时器

555 定时器是一种多用途的单片集成电路。利用它可以很方便地构成施密特触发器、单稳态触发器和多谐振荡器，由于使用灵活、方便，所以 555 定时器在波形的产生与变换、测量与控制、家用电器、电子玩具等许多领域中都得到了广泛的应用。



9.5.1 555 定时器的逻辑功能

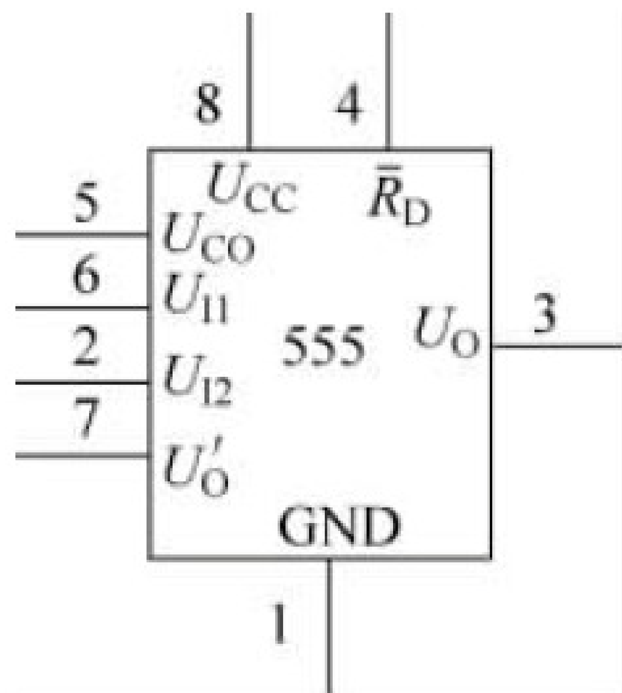


图 9.5.1 555 定时器引脚图

555 定时器的引脚图如图 9.5.1 所示。图中，U_{I1}、U_{I2}为两个外部输入端，U_{CO}为精准比较电压输入端，U_O 为输出端、U'_O为集电极开路的输出端。$\bar{R}_D$ 为复位端。

555 定时器的功能表如表 9.5.1 所示。

表 9.5.1 555 定时器的功能表

U_{I1}	U_{I2}	$\bar{R}_D$	输出 U_O	T_D状态
×	×	0	0	导通
$>\frac{2}{3}U_{CC}$	$>\frac{1}{3}U_{CC}$	1	0	导通
$>\frac{2}{3}U_{CC}$	$<\frac{1}{3}U_{CC}$	1	1	截止
$<\frac{2}{3}U_{CC}$	$>\frac{1}{3}U_{CC}$	1	保持	保持
$<\frac{2}{3}U_{CC}$	$<\frac{1}{3}U_{CC}$	1	1	截止

功能表第 1 行表明，$\bar{R}_D=0$ 时，555 定时器复位，输出为 0，输入无效。功能表第 2～5 行具体说明了 $\bar{R}_D$ 无效时的输入输出关系。当 U_{I1}、U_{I2}均为高电平时，输入输出关系对应功能表第 2 行，输出为 0。集电极开路的三极管 T_D发射结导通。

当然，如表 9.5.1 所示的 555 定时器功能表是由 555 集成定时器的电路结构决定的。如图 9.5.2 所示是 555 集成定时器的电路结构图，包括电阻分压器、比较器、基本 RS 触发器、晶体管开关和输出缓冲器 4 个部分，各部分简要解释如下：

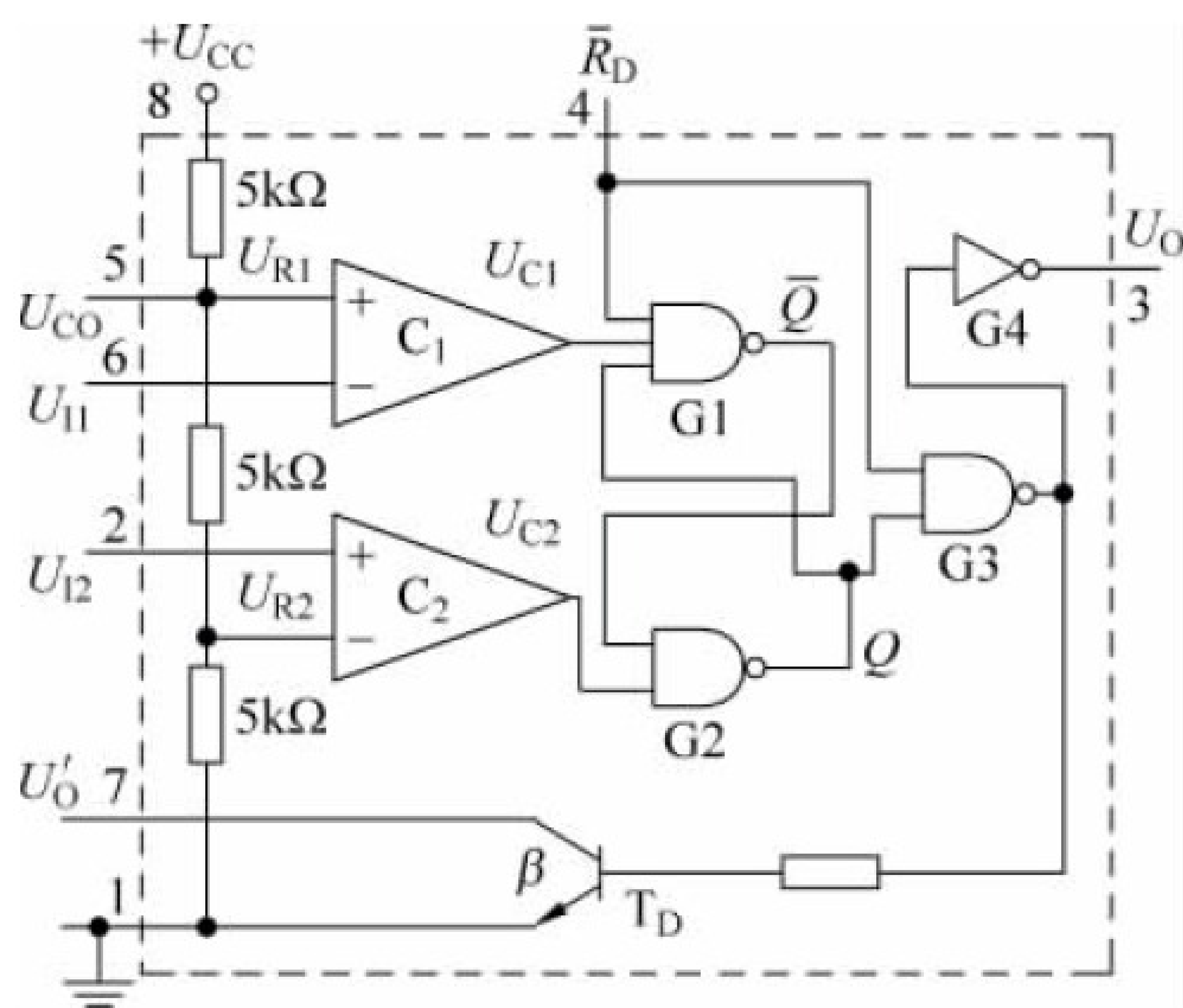


图 9.5.2　555 定时器的电路结构图

1. 比较器

C_1、C_2 为两个电压比较器。用运算放大器做成的比较器有两个电压输入端 U_+、U_-，当 $U_+>U_-$ 时，其输出 U_C 为高电平；当 $U_+<U_-$ 时，其输出 U_C 为低电平。比较器的两个输入端基本上不向外电路索取电流，即输入电阻趋于无穷大。

2. 电阻分压器

电阻分压器由三个电阻值均为 5kΩ 的电阻串联起来组成，555 也因此而得名。电阻分压器为比较器 C_1 和 C_2 提供参考电压。

当电压控制端 U_{CO} 没有外接固定电压时，$U_{R1}=\frac{2}{3}U_{CC}$、$U_{R2}=\frac{1}{3}U_{CC}$。如果电压控制端 U_{CO} 外接固定电压，则 C_1 和 C_2 的参考电压为 $U_{R1}=U_{CO}$ 和 $U_{R2}=\frac{1}{2}U_{CO}$。当在工作中不使用 U_{CO} 端时，一般都通过一个 0.01μF 的电容接地，以防止高频干扰。

3. 基本 RS 触发器

与非门 G1、G2 组成基本 RS 触发器。$\bar{R}_D$ 端是专门设置的可从外部进行置 0 的复位端，当 $\bar{R}_D=0$ 时，输出 $U_O=0$。正常工作时 $\bar{R}_D=1$。

4. 晶体管开关和输出缓冲器

晶体管 T_D 构成开关，其状态受 $\bar{Q}$ 端控制，当 $\bar{Q}=1$ 时，T_D 导通。反相器 G4 为输出缓冲器，它的作用是提高 555 定时器的带负载能力和隔离负载对定时器的影响。

可由如图 9.5.2 所示电路进一步求出 555 定时器的功能表，具体如下：

当 $\bar{R}_D=0$ 时，$\bar{Q}=1$，输出电压 $U_O=Q=0$，三极管 T_D 发射结导通，如表 9.5.1 第 1 行所示。

当 $\bar{R}_D=1$，$U_{I1}>U_{R1}$，$U_{I2}>U_{R2}$ 时，比较器 C_1 的输出 $U_{C1}=0$，比较器 C_2 的输出 $U_{C2}=1$，基本 RS 触发器置 0，输出 $U_O=Q=0$，同时 T_D 导通，如表 9.5.1 第 2 行所示。

当 $\bar{R}_D=1$，$U_{I1}>U_{R1}$，$U_{I2}<U_{R2}$ 时，比较器 C_1 的输出 $U_{C1}=0$，比较器 C_2 的输出 $U_{C2}=0$，基本触发器为 $Q=\bar{Q}=1$，则输出 $U_O=Q=1$。如表 9.5.1 第 3 行所示。

当 $\bar{R}_D=1$，$U_{I1}<U_{R1}=\frac{2}{3}U_{CC}$，$U_{I2}>U_{R2}=\frac{1}{3}U_{CC}$ 时，比较器 C_1 的输出 $U_{C1}=1$，比较器 C_2 的输出 $U_{C2}=1$，基本 RS 触发器保持原来状态不变。所以，输出电压 U_O 的状态也维持不变，如表 9.5.1 第 4 行所示。

当 $\bar{R}_D=1$，$U_{I1}<U_{R1}$，$U_{I2}<U_{R2}$ 时，比较器 C_1 的输出 $U_{C1}=1$，比较器 C_2 的输出 $U_{C2}=0$，基本 RS 触发器置 1，输出 $U_O=Q=1$，同时 T_D 截止，如表 9.5.1 第 5 行所示。

9.5.2 将 555 定时器接成施密特触发器



555 定时器应用十分广泛，利用 555 定时器可以很方便地构成施密特触发器。

当然，555 定时器具有 2 个输入、1 个输出，施密特触发器只具有 1 个输入、1 个输出，要用 555 定时器实现施密特触发器，应设法将 555 的 2 个输入变成 1 个输入，且具有施密特触发器的特性即可。

将 2 个输入变为 1 个输入，主要有 3 种思路：

(1) 输入：U_{I1}，U_{I2} 接固定电平。

(2) 输入：U_{I2}，U_{I1} 接固定电平。

(3) 输入：U_{I2}、U_{I1} 短接。

思路(1)的实现方法下，若 U_{I2} 接 1，有 $U_{I2}>1/3U_{CC}$。查表 9.5.1 第 2、4 行，有 $U_O=0$，无两个稳态，不具有施密特触发器的特性。若 U_{I2} 接 0，有 $U_{I2}<1/3U_{CC}$。查功能表第 3、5 行，有 $U_O=1$，无两个稳态，也不具有施密特触发器的特性。

思路(2)的实现方法下，若 U_{I1} 接 1，有 $U_{I1}>2/3U_{CC}$。查功能表第 2、3 行，有两个稳态，但 $U_{T+}=U_{T-}=1/3U_{CC}$，不具有施密特触发器的特性。若 U_{I1} 接 0，有 $U_{I1}<2/3U_{CC}$。查功能表第 4、5 行，有 $U_O=1$，无两个稳态，不具有施密特触发器的特性。

可采用思路(3)实现施密特触发器，参考电路如图 9.5.3 所示。图中将两个比较器的输入端 U_{I1}、U_{I2} 连在一起，作为施密特触发器的输入端 U_I，U_{CO} 端接有 0.01μF 的电容，主要起滤波作用，以提高比较器参考电压的稳定性；$\bar{R}_D$ 端接至电源 U_{CC}，以提高可靠性。

在如图 9.5.3 所示电路的输入 U_I 端加上如图 9.5.4(a)所示的三角波，电路的输出 U_O 如图 9.5.4(b)所示。

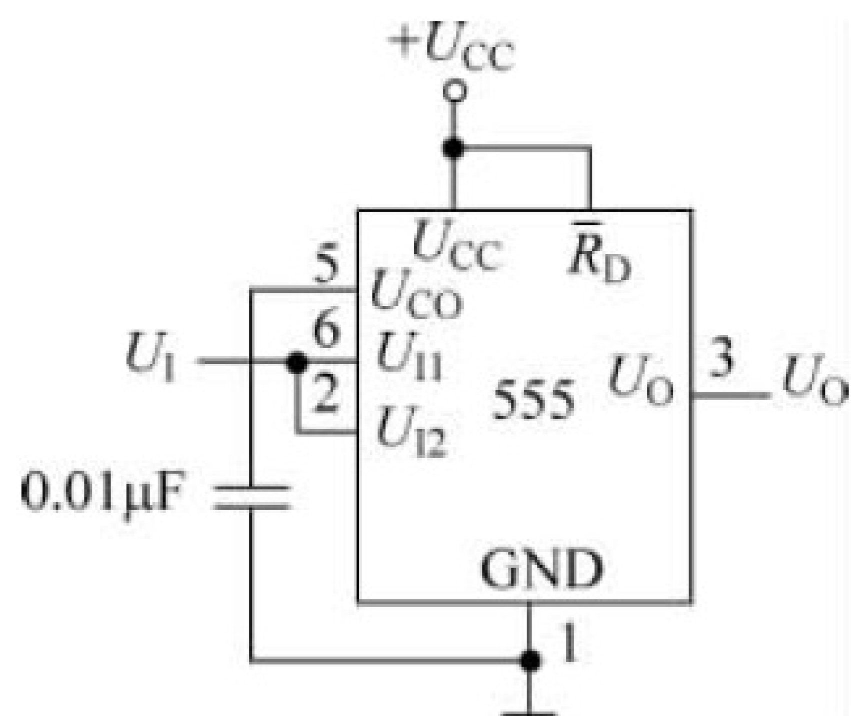


图 9.5.3　电路图

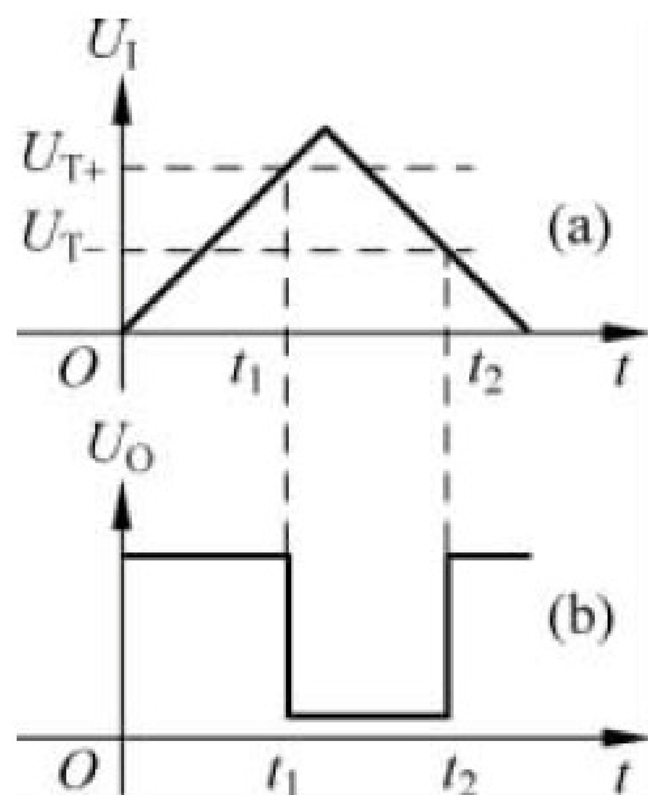


图 9.5.4　工作波形

可结合表 9.5.1 来理解工作波形。

(1) $0 \leqslant t < t_1$。

$t=0$ 时，由于 $U_{I1}=U_{I2}=U_I=0V$，由表 9.5.1 第 5 行，输出 $U_O=1$。

在 t_1时刻以前，U_I虽然在上升，但由表 9.5.1，当 $U_I<\frac{2}{3}U_{CC}$时，输出 $U_O=1$。

(2) $t_1 \leqslant t < t_2$。t_1时刻 U_I增加到$\frac{2}{3}U_{CC}$，则当 $t=t_{1+}$时有

$$U_{I1}=U_{I2}=U_I>\frac{2}{3}U_{CC}$$

由表 9.5.1 第 2 行，输出 $U_O=0$。

$t>t_{1+}$后，虽然输入 U_I先上升，达到最大值后开始减小，但在 t_2时刻以前，$U_I>\frac{1}{3}U_{CC}$，电路的输出 $U_O=0$。

(3) $t_2 \leqslant t$。t_2时刻 U_I减小到$\frac{1}{3}U_{CC}$，则当 $t=t_{2+}$时有

$$U_{I1}=U_{I2}=U_I<\frac{1}{3}U_{CC}$$

由表 9.5.1 第 5 行，输出 $U_O=1$，电路又回到初始稳态。

通过上述分析，可得出，图 9.5.3 为用 555 定时器接成的施密特触发器。

通过上述分析，可得出接通电位 $U_{T+}=\frac{2}{3}U_{CC}$，断开电位 $U_{T-}=\frac{1}{3}U_{CC}$，故回电压为

$$\Delta U_T=U_{T+}-U_{T-}=\frac{2}{3}U_{CC}-\frac{1}{3}U_{CC}=\frac{1}{3}U_{CC} \tag{9.5.1}$$

从图 9.5.4 的波形图还可以看出，施密特触发器可以用来进行波形变换。这个例子就是将三角波变换为方波。

9.5.3　将 555 定时器接成单稳态触发器



利用 555 定时器还可以很方便地构成单稳态触发器，用 555 定时器接成的单稳态触发器如图 9.5.5 所示。图中，R、C 为定时元件，U_I为输入触发器信号，接在 U_{I2}(2)端，当 U_I的下降沿到来时，触发器触发。(7)端和(6)U_{I1}端短

接，$\overline{R}_D$ 端不用，接 U_{CC}。

稳态下，$U_I = U_{I2} = 1$，有 $U_{I2} > 1/3U_{CC}$。查功能表第 2、4 行，有 $U_O = 0$、$U'_O = 0$。由于 $U'_O = 0$，有 $U_{I1} = 0$，电容 C 上没有储存电荷。

当然，当 $U_I = U_{I2} = 1$ 时，若 U_O 的状态为 1，电容 C 将被充电，U_{I1} 电位将上升，当 $U_{I1} = 2/3U_{CC}$ 时，查功能表第 2 行，有 $U_O = 0$。之后，电容 C 放电，回到初始稳定状态。

当 U_I 跳变到 0 时，有 $U_I = U_{I2} = 0$，$U_{I1} = 0$，查功能表第 5 行，有 $U_O = 1$。555 定时器进入暂稳态。之后，负脉冲撤除，有 $U_I = U_{I2} = 1$、$U_{I1} = 0$，查功能表第 4 行，保持 $U_O = 1$ 不变。由于 U_O 的状态为 1，电容 C 将被充电，当 $U_{I1} = 2/3U_{CC}$ 时，555 定时器翻转到 0，进入稳定状态。之后，电容 C 放电，回到初始稳定状态。

在如图 9.5.5 所示电路的输入 U_I 端加上如图 9.5.6(a)所示的脉冲，电路的输出 U_O 如图 9.5.6(b)。没有触发信号时，U_I 为高电平，当电源接通后，电路自动稳定在 $U_O = 0$ 的稳态。当触发脉冲 U_I 的下降沿到来时，电路被触发，触发器置 1，进入暂稳态，经过一段时间，电路又自动返回到稳态。

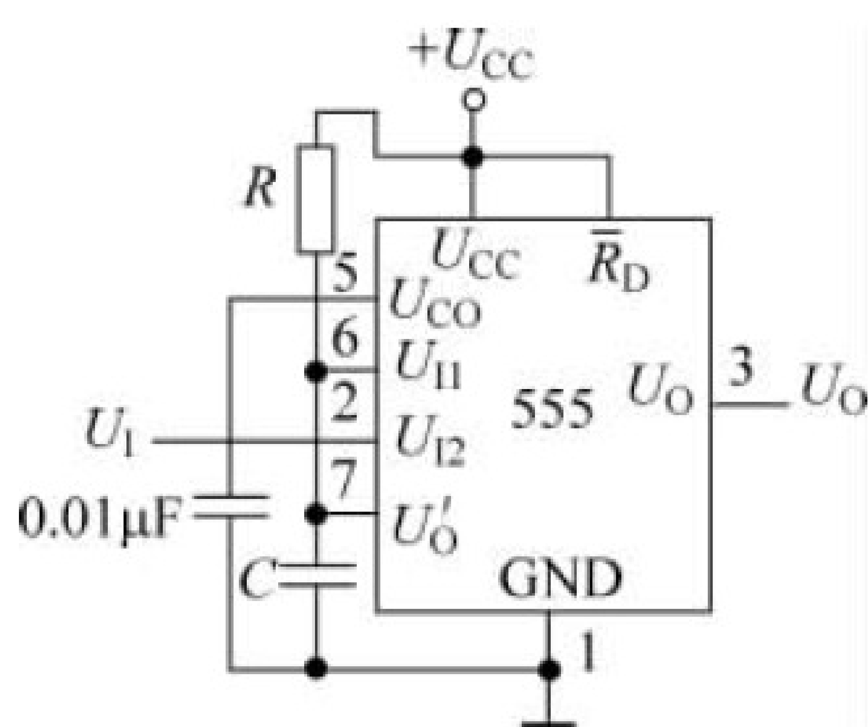


图 9.5.5 电路图

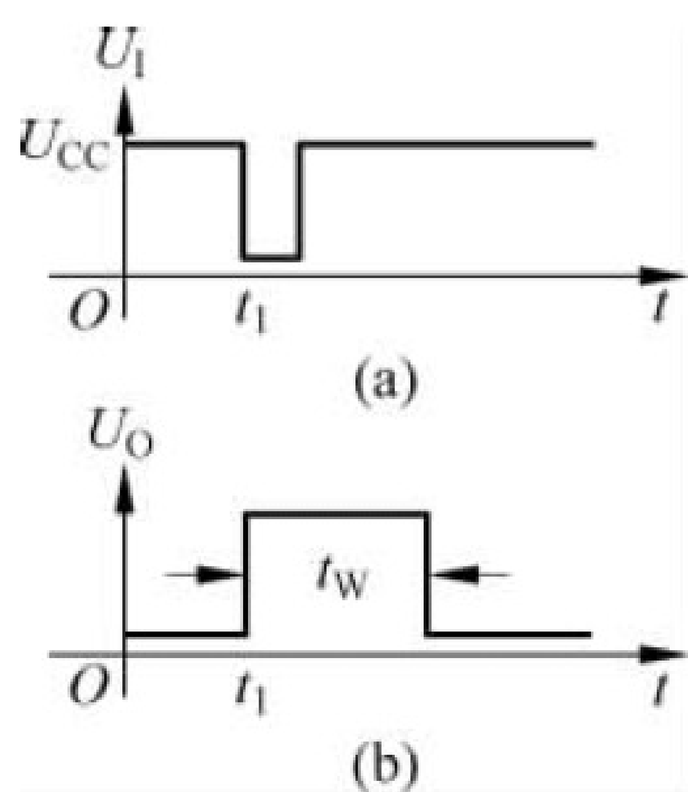


图 9.5.6 工作波形

通过上面的分析，可知暂稳态持续时间为电容 C 从 0 上升到 $2/3U_{CC}$ 的充电时间，由此可求出输出脉冲 U_O 的脉冲宽度

$$t_w = RC\ln\frac{U_{CC} - 0}{U_{CC} - \frac{2}{3}U_{CC}} = RC\ln 3 \approx 1.1RC \tag{9.5.2}$$

通常，电阻 R 的取值为几百欧姆到几兆欧姆，电容的取值范围为几百皮法到几百微法，故脉冲宽度可在几微秒到几分钟内调节。



9.5.4 将 555 定时器接成多谐振荡器

9.4.2 节介绍了用施密特触发器组成多谐振荡器的方法，参考电路如图 9.4.4 所示。9.5.2 节介绍了用 555 定时器构成施密特触发器的方法，参考电路如图 9.5.3 所示。

可基于如图 9.5.3 所示的施密特触发器进一步构成多谐振荡器，参考电路如图 9.5.7 所示。图中，555 定时器的 U_{I1}、U_{I2} 短接构成施密特触发器。U_{I1}、U_{I2} 短接后通过电容 C 接

地,通过电阻 R_2 与输出 U_O 连接。参考如图 9.4.4 所示电路,如图 9.5.7 所示电路为多谐振荡器。

读者不要忘记,555 定时器还具有一个集电极开路的输出端 U'_O。当 U'_O 通过上拉电阻与电源接通后,与输出 U_O 具有相同的逻辑状态。利用 U'_O 实现的多谐振荡器电路如图 9.5.8 所示。

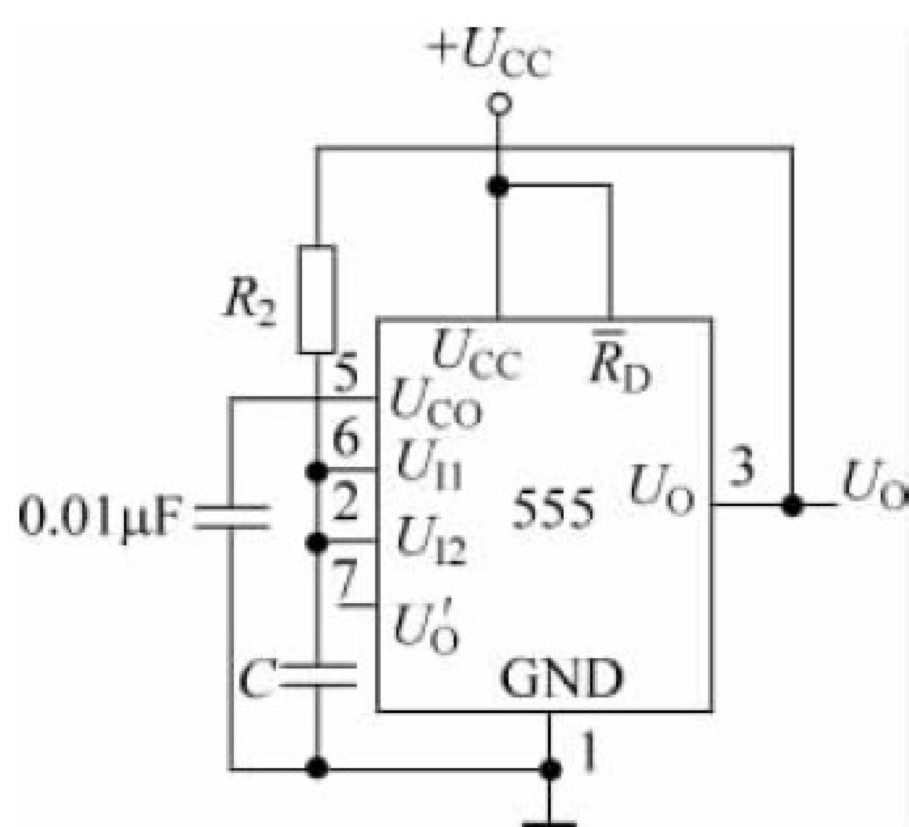


图 9.5.7　多谐振荡器电路

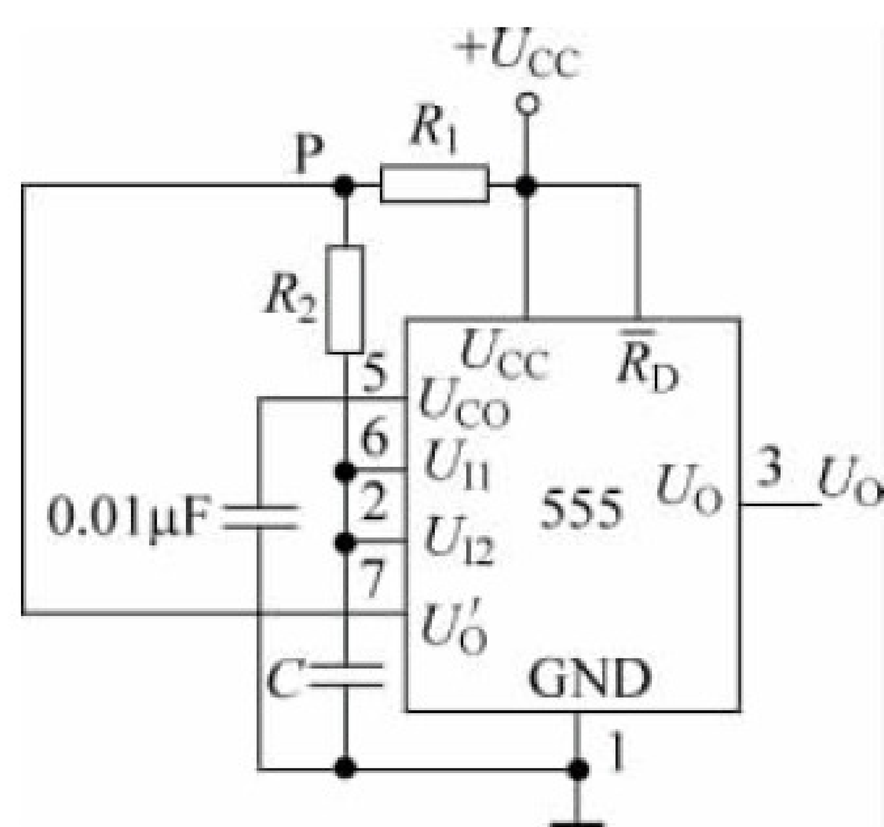


图 9.5.8　改进的多谐振荡器电路

如图 9.5.9 所示为电路中 U_O 的输出波形。简要分析如下:

当电源接通时,电容 C 还未充电,所以 $U_{I1}=U_{I2}=0$,由表 9.5.1 第 5 行,输出 $U_O=1$,三极管 T_D 截止,$U'_O=1$。$U'_O=1$,将对电容 C 充电,电容上的电压 U_C 增加,U_{I1}、U_{I2} 电压也增加,当 $U_{I1}=U_{I2}=U_C\geqslant\frac{2}{3}U_{CC}$ 时,由表 9.5.1 第 2 行,输出 $U_O=0$,三极管 T_D 饱和导通,$U'_O=0$。而此时电容上的电压为 $\frac{2}{3}U_{CC}$,将通过电阻 R_2 放电,电容上的电压 U_C 下降,U_{I1}、U_{I2} 电压也下降,当 $U_{I1}=U_{I2}=U_C\leqslant\frac{1}{3}U_{CC}$ 时,由表 9.5.1 第 5 行,输出 $U_O=1$,周而复始,输出如图 9.5.9 所示的矩形波。

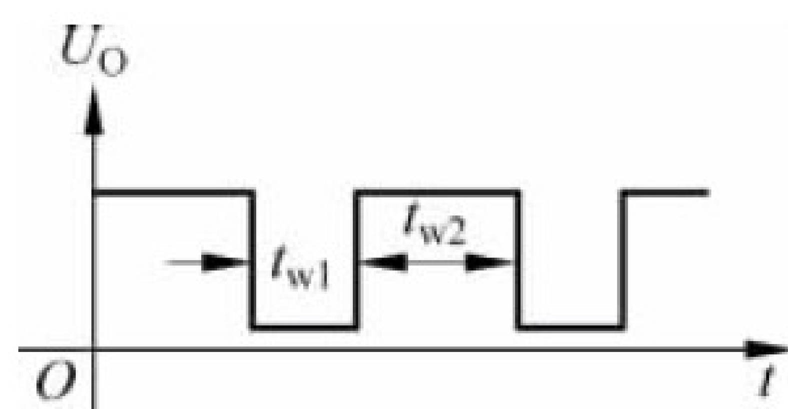


图 9.5.9　图 9.5.8 的输出波形

下面不加证明地给出输出矩形波 U_O 的脉冲宽度 t_{w1}、t_{w2} 及周期 T 的计算公式:

$$t_{w1}=0.7R_2C \tag{9.5.3}$$

$$t_{w2}=0.7(R_1+R_2)C \tag{9.5.4}$$

$$T=t_{w1}+t_{w2}=0.7(R_1+2R_2)C \tag{9.5.5}$$

9.5.5　应用实例

1. 用 555 电路组成模拟声响电路

如图 9.5.10 所示电路,是用两个 555 电路分别接成两个多谐振荡器构成的模拟声响电路。在多谐振荡器 1 中,调节定时元件 R_{11}、R_{12}、C_1,使多谐振荡器 1 的输出信号 U_{O1} 的频率为 $f_{O1}=1\text{Hz}$。在多谐振荡器 2 中,调节定时元件 R_{21}、R_{22}、C_2 使多谐振荡器 2 的输出信号

U_{O2}的频率为 $f_{O2}=1\text{kHz}$。U_{O1}、U_{O2}的输出波形如图 9.5.11 所示。

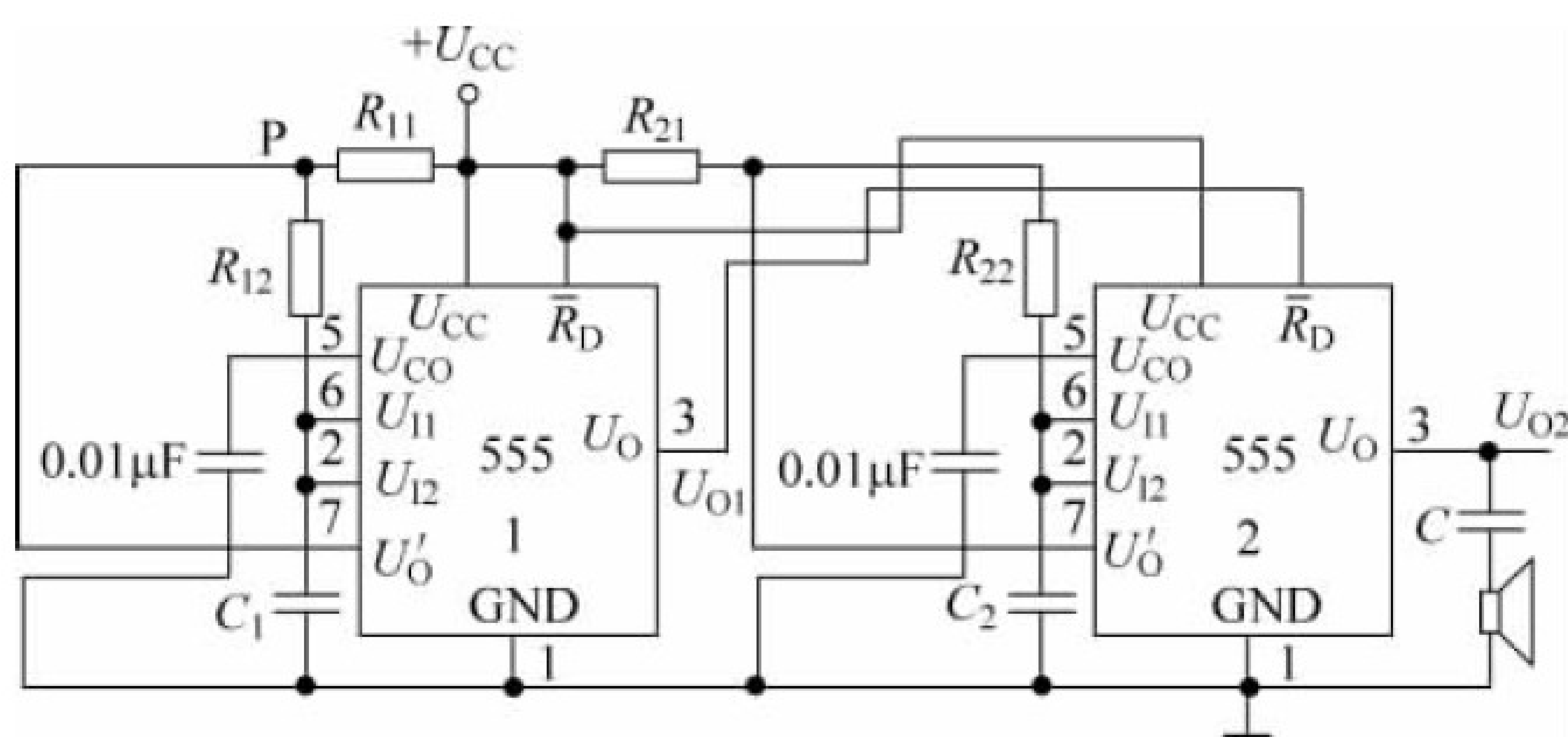


图 9.5.10 用 555 定时器组成模拟声响电路

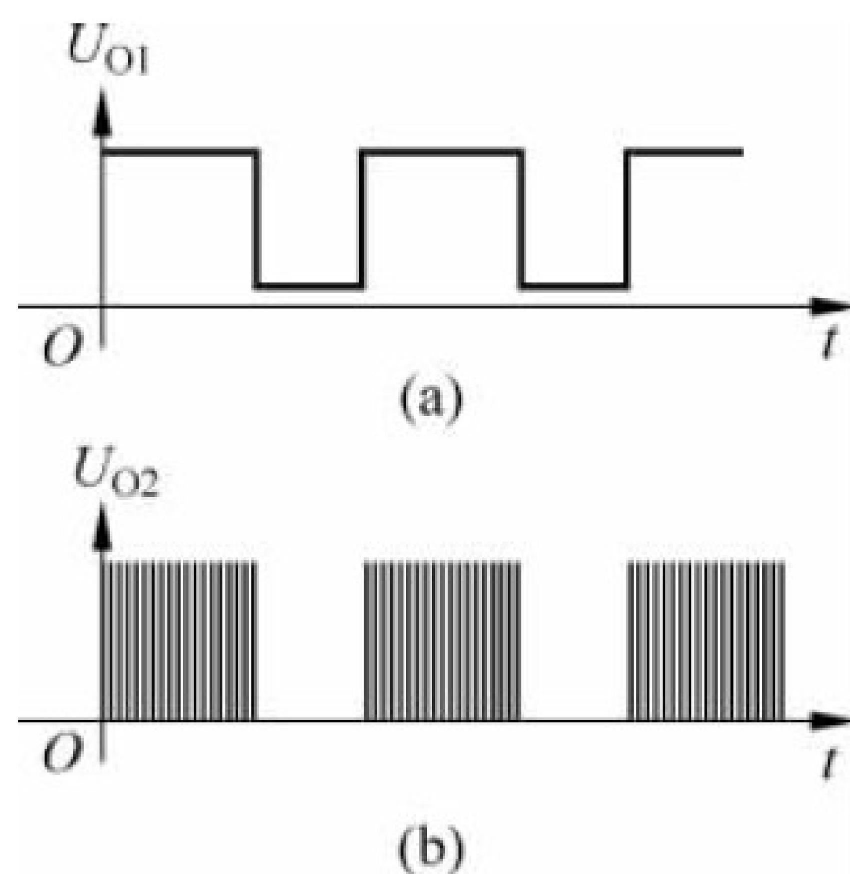


图 9.5.11 图 9.5.10 的工作波

多谐振荡器 1 的输出 U_{O1}接至多谐振荡器 2 的 $\bar{R}_D$ 端(4 脚)，所以当 U_{O1}为低电平时，第 2 片 555 定时器处于复位状态，$U_{O2}=0$，多谐振荡器 2 停止振荡。人耳的听觉范围为20Hz～20kHz，1kHz 的电压信号加在扬声器上将使扬声器发声，因此，如图 9.5.10 所示电路将会使扬声器发出“呜呜——”的间隙声响。

2. 用 555 电路组成失落脉冲检出电路

如图 9.5.12 所示电路为失落脉冲检出电路。它是在用 555 电路接成单稳态触发器的基础上，在定时电容 C 的两端接一个三极管 T，此三极管 T 的基极接至输入信号 U_I，其工作原理如下：

在图 9.5.12 中，调节时间常数 RC，使被监视的输入信号 U_I，在正常频率下电容器上的电压充不到触发电压$\frac{2}{3}U_{CC}$，这就使 555 电路的输出 $U_O=1$。当被监视的信号 U_I的频率 f_I 不正常时(如频率降低或中间失掉一个脉冲)，则电容器上的电压 U_C将充到$\frac{2}{3}U_{CC}$，使 555 电路的输出为 $U_O=0$；如果输入信号 U_I的频率恢复正常后，输出电压 U_O又恢复到高电平。其工作波形如图 9.2.13 所示。

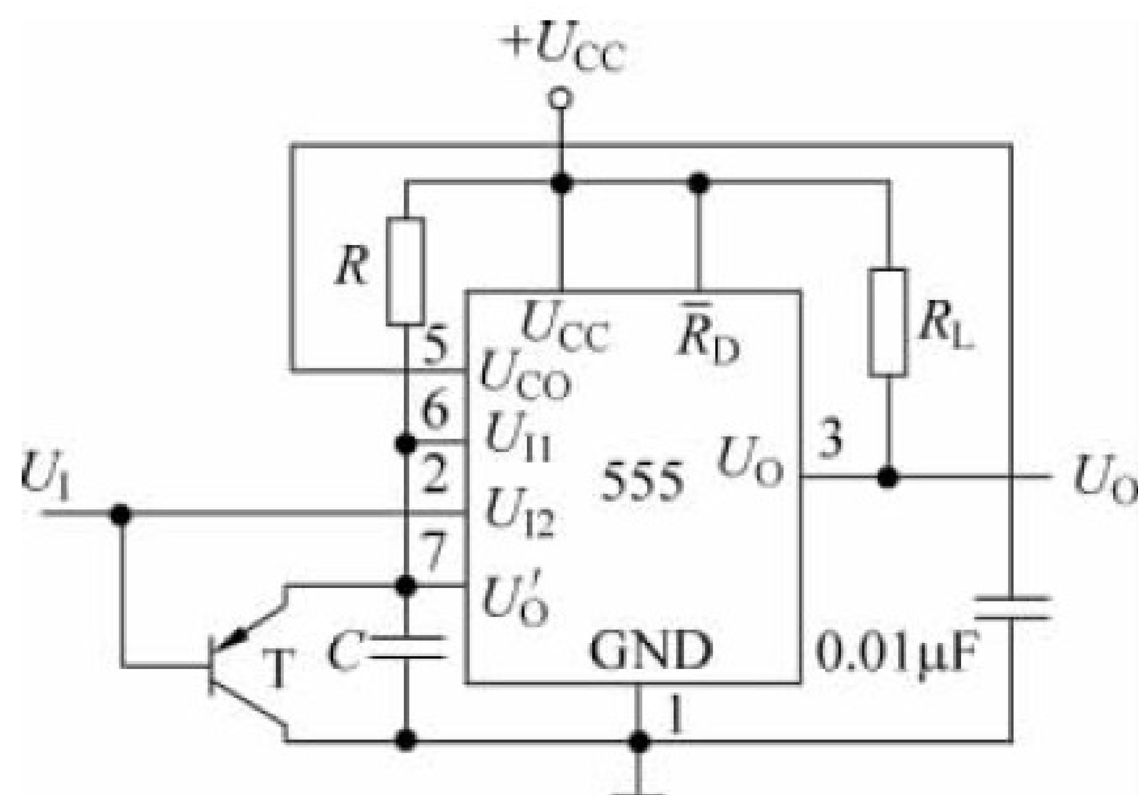


图 9.5.12 用 555 电路组成失落脉冲检出电路

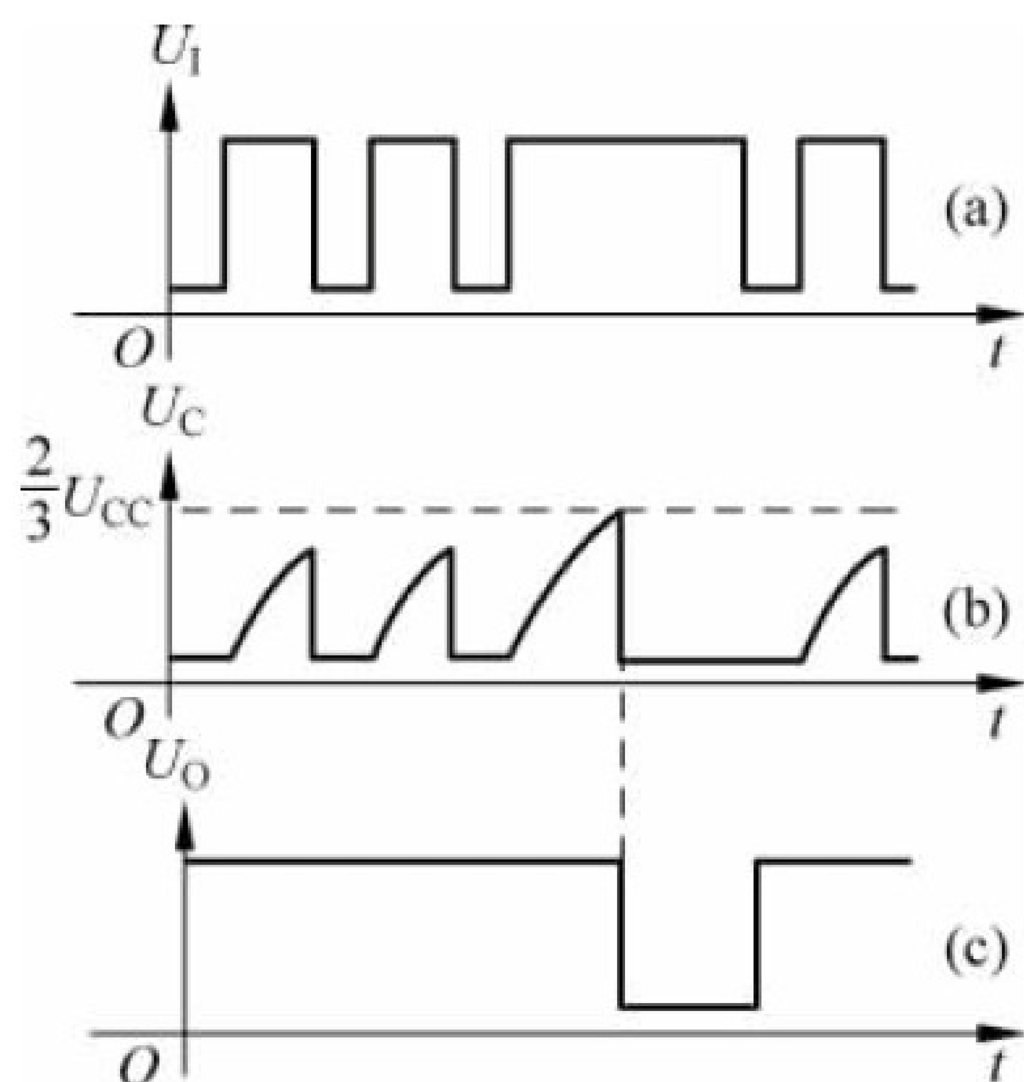


图 9.5.13 图 9.5.12 的工作波形

利用这个原理，可以对机器的转速或者人体的心率进行监视，机器的转速或者人体的心率通过传感器，转换成模拟信号，将此模拟信号接至图 9.5.12 电路的输入端 U_I，当机器的转速降到一定的限度或者人体的心率不齐时，就会发生报警。

复习与思考

9.5.1　说出 555 定时器得名的由来。

9.5.2　如图 9.2.3 所示的用门电路组成的施密特触发器是通过正反馈保证输出波形的性能的，那么 555 定时器通过什么方式保证输出波形的性能？

习题

9-1　填空题

1. 施密特触发器具有________稳态，而且每个稳态都需要________信号才能维持。施密特触发器的一个重要参数是________电压。

2. 单稳态触发器具有一个________、一个________。单稳态触发器的________维持时间的长短只取决于________，而与触发脉冲的幅度和宽度无关。

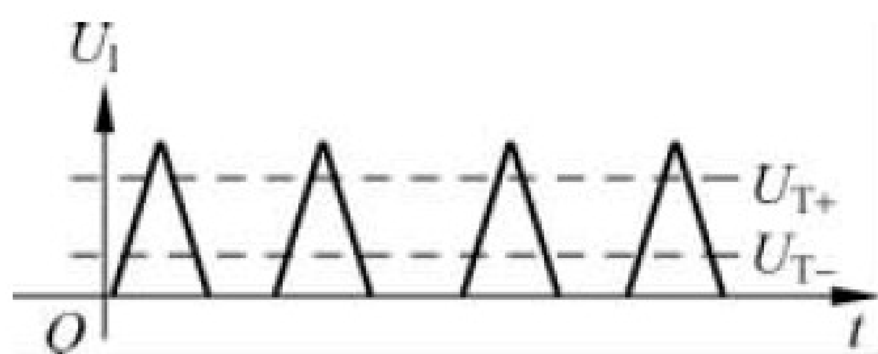
3. 多谐振荡器是一种能够产生________和________的矩形波的电路。它不需要外加输入信号的作用，它没有________，所以又称为________电路。多谐振荡器可产生________信号，是绝大多数数字系统的必备电路。

9-2　分析应用题

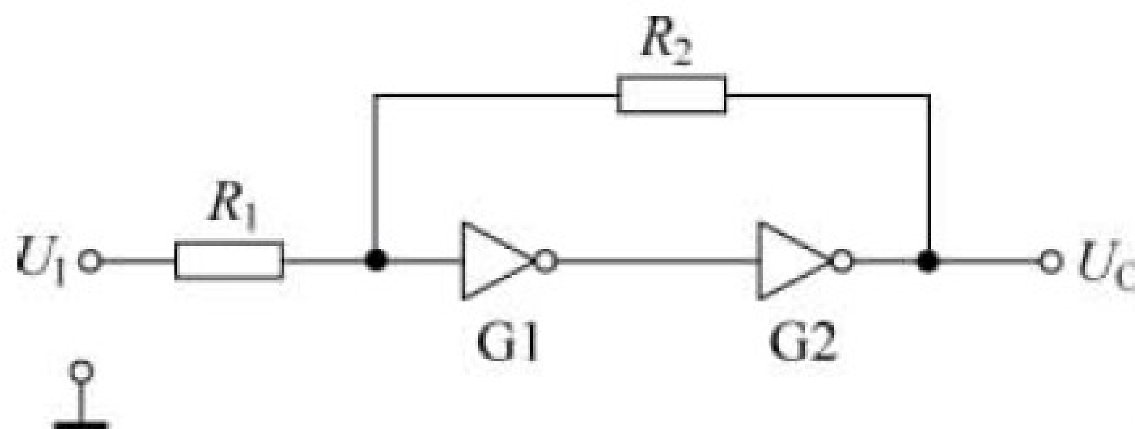
1. 如题图 9.1 所示输入信号加在施密特反相器的输入端，画出输出信号的波形。施密特反相器的 U_{T+}、U_{T-} 已在图中标出。

2. 如图 9.2.3 所示用 CMOS 门组成的施密特触发器中，$R_1=30\text{k}\Omega$，$R_2=60\text{k}\Omega$，$U_{DD}=10\text{V}$，$U_{TH}=5\text{V}$，求回差电压 ΔU_T。

3. 如题图 9.2 所示电路中的门电路为 CMOS 门电路，$R_1=10\text{k}\Omega$，$R_2=30\text{k}\Omega$，$U_{DD}=12\text{V}$，求 U_{T+}、U_{T-}。



题图　9.1



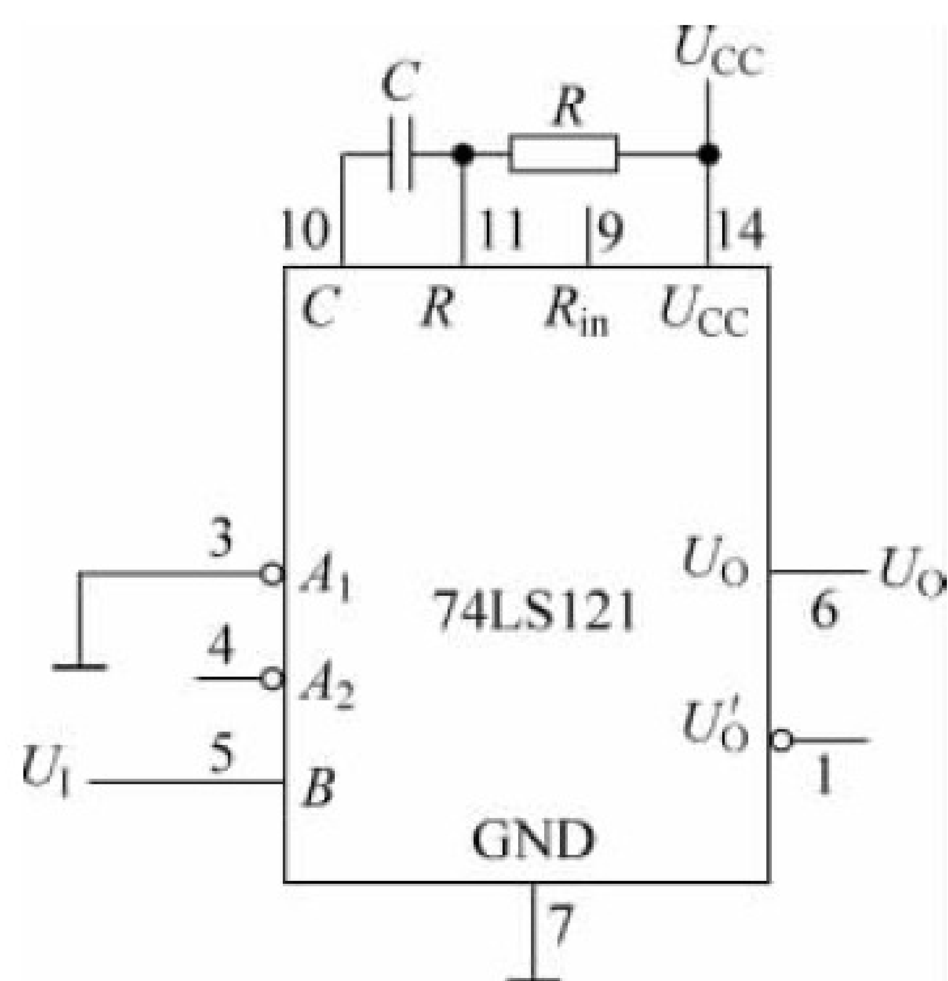
题图　9.2

4. 如图 9.2.3 所示电路中，如果要求接通电位 $U_{T+}=7.5\text{V}$，回差电压 $\Delta U_T=5\text{V}$，求 R_1、R_2 及 U_{DD} 的值。

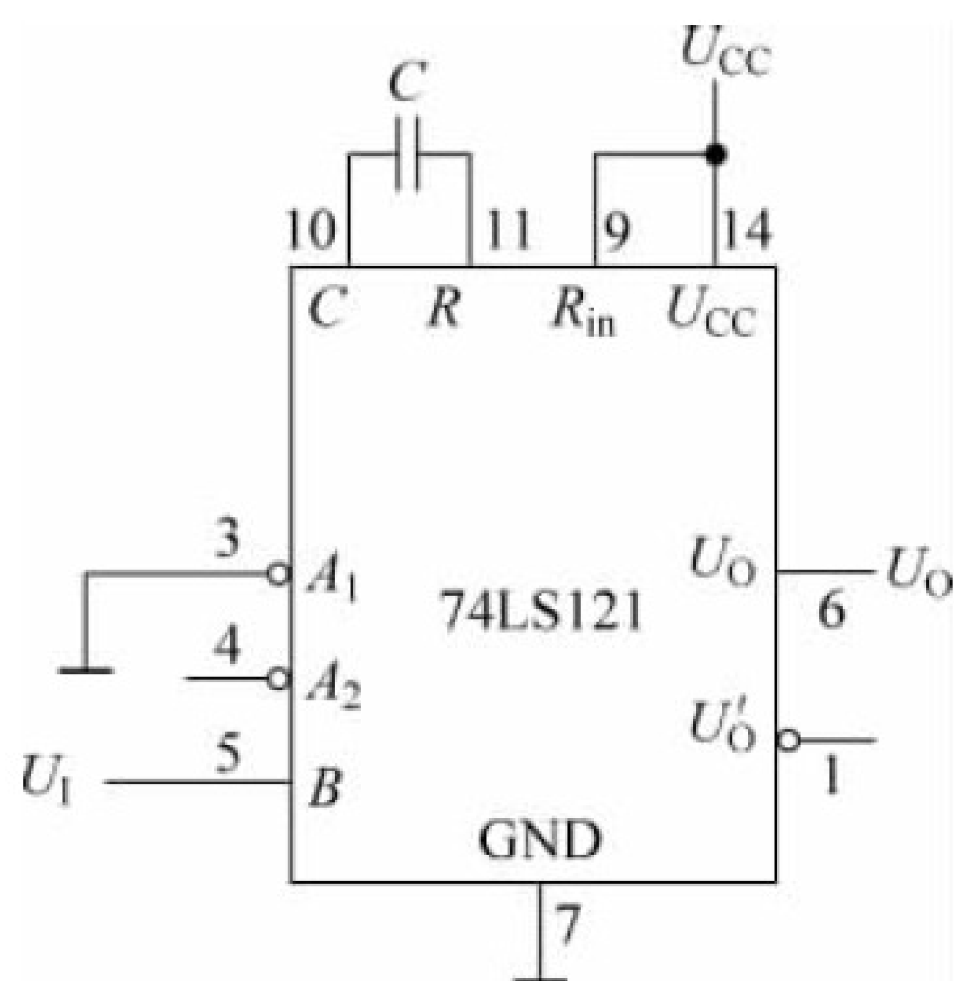
5. 如图 9.3.1 所示为用 CMOS 门电路构成的微分型单稳态触发器，$R_I=R=33\text{k}\Omega$，$C=0.1\mu\text{F}$，$U_{DD}=12\text{V}$，求在触发脉冲作用下输出脉冲的宽度与幅度。

6. 如题图 9.3 所示单稳态触发器中 $R=33\text{k}\Omega$，$C=0.01\mu\text{F}$，$U_{\text{CC}}=5\text{V}$，求在触发脉冲作用下输出脉冲的宽度。

7. 如题图 9.4 所示单稳态触发器中内部电阻 $R_{\text{in}}=2\text{k}\Omega$，$C=0.1\mu\text{F}$，$U_{\text{CC}}=5\text{V}$，请求在触发脉冲作用下输出脉冲的宽度。

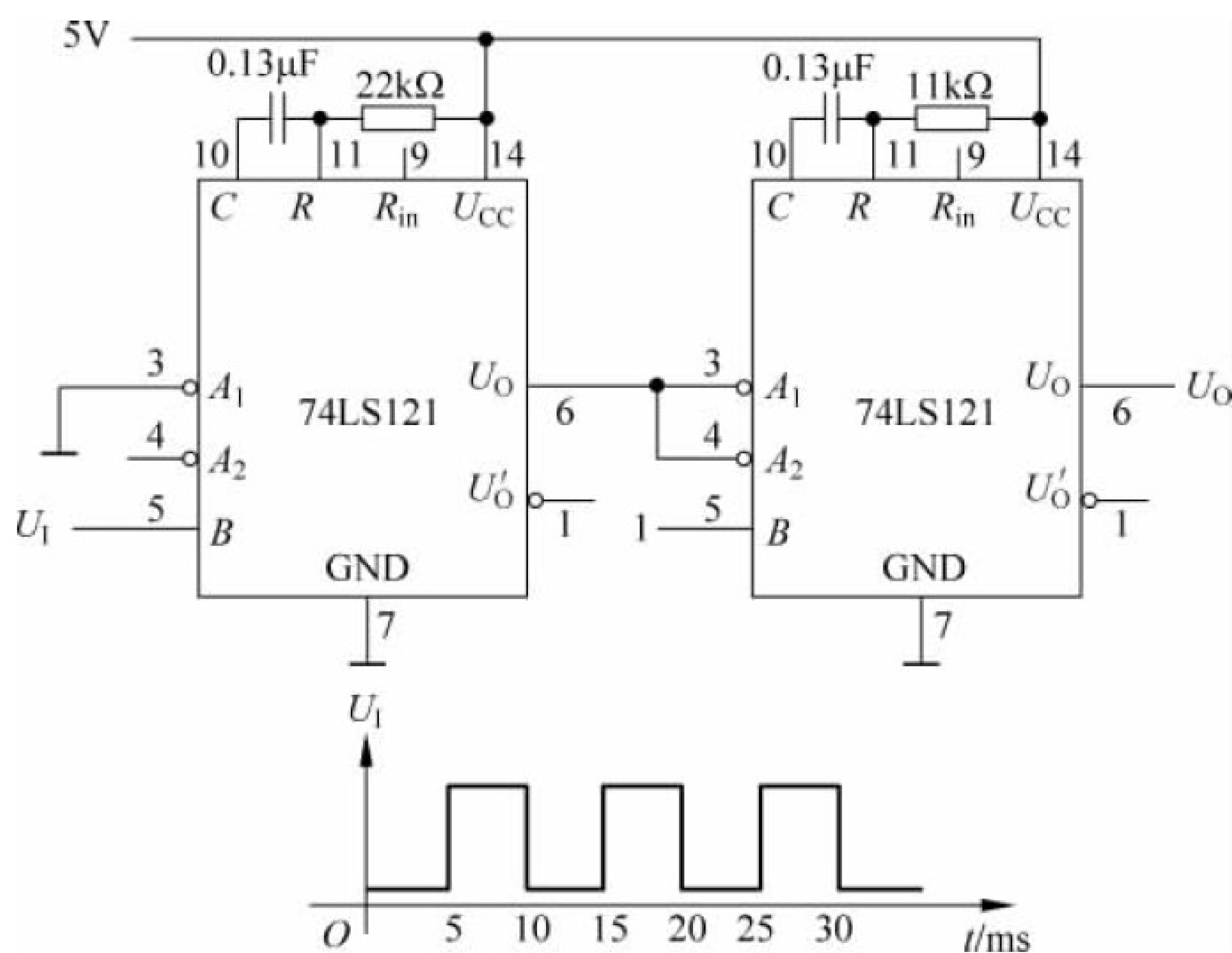


题图 9.3



题图 9.4

8. 画出如题图 9.5 所示脉冲变换电路在输入脉冲作用下相应的输出 U_{O} 的波形。



题图 9.5

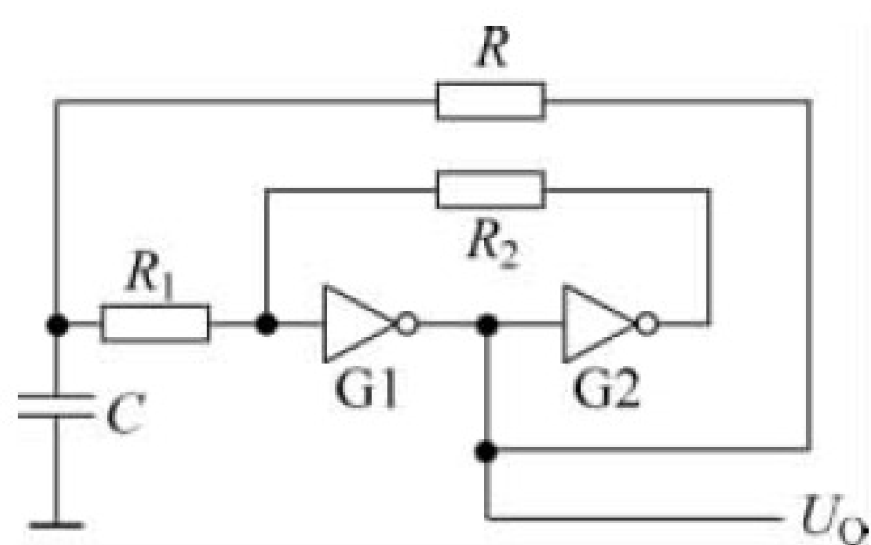
9. 如图 9.4.2 所示非对称式多谐振荡器电路上电时有微弱的正变化，画出电路的反馈变化过程。

10. 在如图 9.4.4 所示电路中，$R=30\text{k}\Omega$，$C=0.1\mu\text{F}$；施密特触发器的 $U_{\text{T+}}=7\text{V}$，$U_{\text{T-}}=3\text{V}$，$U_{\text{DD}}=10\text{V}$，求该电路的振荡频率。

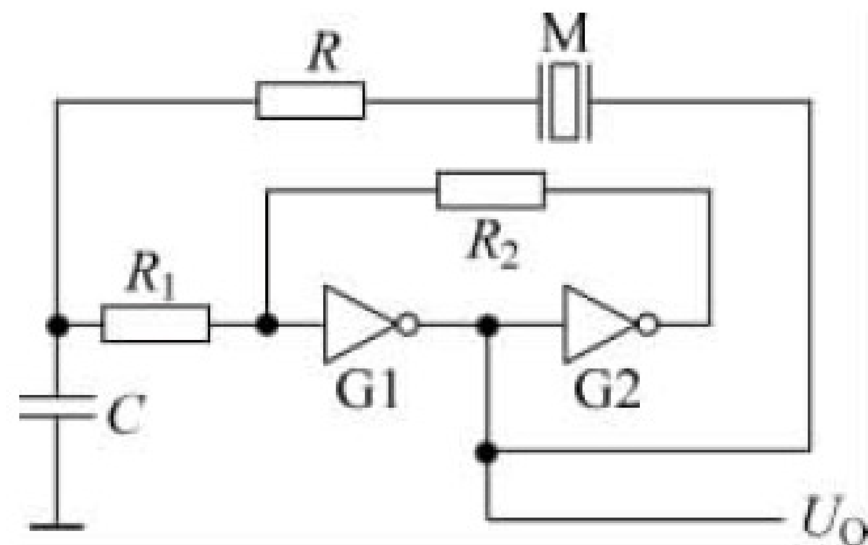
11. 如题图 9.6 所示为用 CMOS 门组成的多谐振荡器电路，图中，$R_1=R=30\text{k}\Omega$，$R_2=60\text{k}\Omega$，$U_{\text{DD}}=10\text{V}$，$U_{\text{TH}}=5\text{V}$，$C=0.1\mu\text{F}$，求该电路 U_{O} 的输出频率。

12. 改进如题图 9.7 所示的多谐振荡器电路，实现一个占空比可调的多谐振荡器并画

出原理电路。



题图 9.6

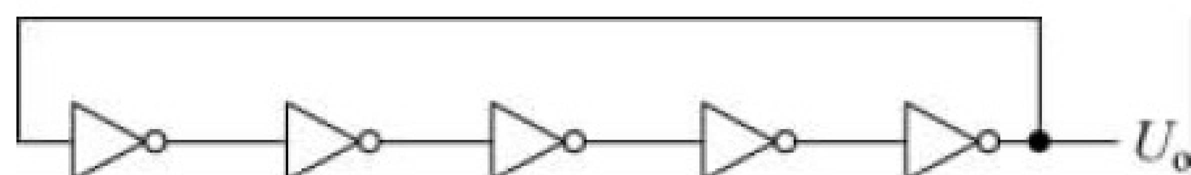


题图 9.7

13. 在如题图 9.7 所示电路中，$R_1=R=30\text{k}\Omega$，$R_2=60\text{k}\Omega$，$U_{DD}=10\text{V}$，$U_{TH}=5\text{V}$。今测得 U_O 的输出频率为 5kHz，求该电路中的电容 C。

14. 在如题图 9.8 所示电路中，$R_1=R=30\text{k}\Omega$，$R_2=60\text{k}\Omega$，$U_{DD}=10\text{V}$，$U_{TH}=5\text{V}$，$C=0.1\mu\text{F}$，晶振 M 的谐振频率为 5M，求该电路的振荡频率。

15. 如题图 9.8 所示电路中 U_O 的输出频率为 20MHz，假定每个反相器的传输延迟时间相同，请求各反相器的平均传输延迟时间。



题图 9.8

16. 如图 9.5.3 所示电路中，$U_{CC}=10\text{V}$，求 U_{T+}、U_{T-}。

17. 如图 9.5.3 所示电路中，$U_{CC}=10\text{V}$，U_{CO}接 6V 的精准电压源，求 U_{T+}、U_{T-}。

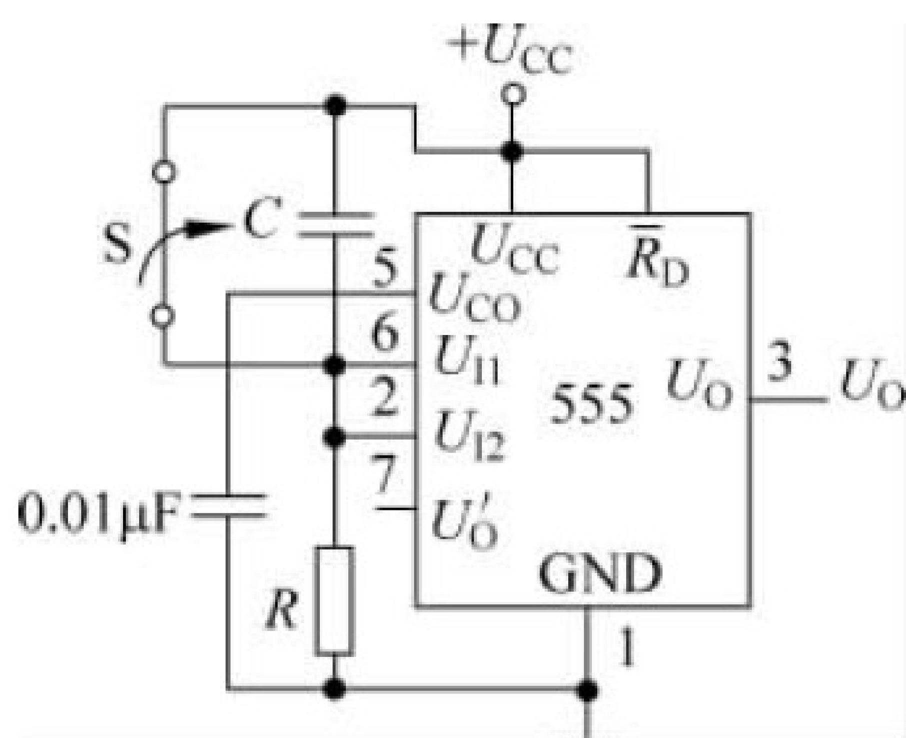
18. 试分析如题图 9.9 所示开机延时电路的工作原理。若 $C=25\mu\text{F}$，$R=19\text{k}\Omega$，$U_{CC}=12\text{V}$，问常闭开关 S 断开以后经过多长时间，输出 U_O 为高电平。

19. 如图 9.5.5 所示单稳态触发器的工作电源 $U_{CC}=12\text{V}$，触发信号来自 TTL 门电路的输出，该单稳态触发器触发器能否正常工作？若不能，可如何改进电路？

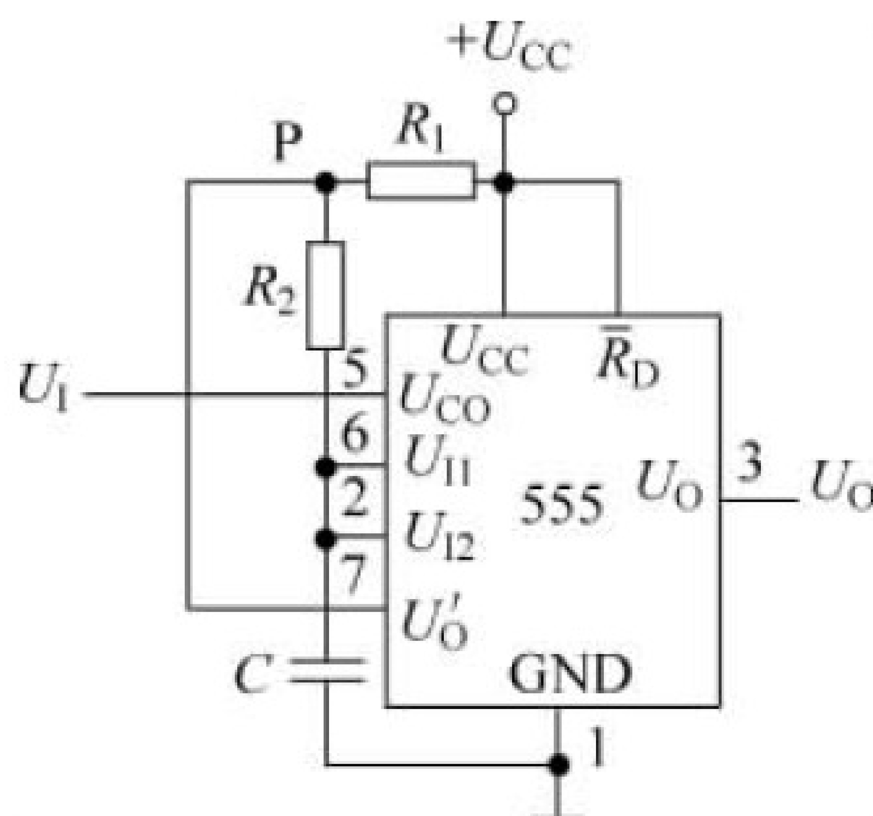
20. 参考图 9.5.5 设计一个触发脉冲来自 TTL 门电路的输出信号，输出脉冲宽度在 1～5s 间可调的单稳态触发器。

21. 在如图 9.5.8 所示电路中，$R_1=R_2=33\text{k}\Omega$，$C=0.1\mu\text{F}$，求该电路的振荡频率。

22. 试分析如题图 9.10 所示电路中输入电压 U_I和振荡频率之间的关系。

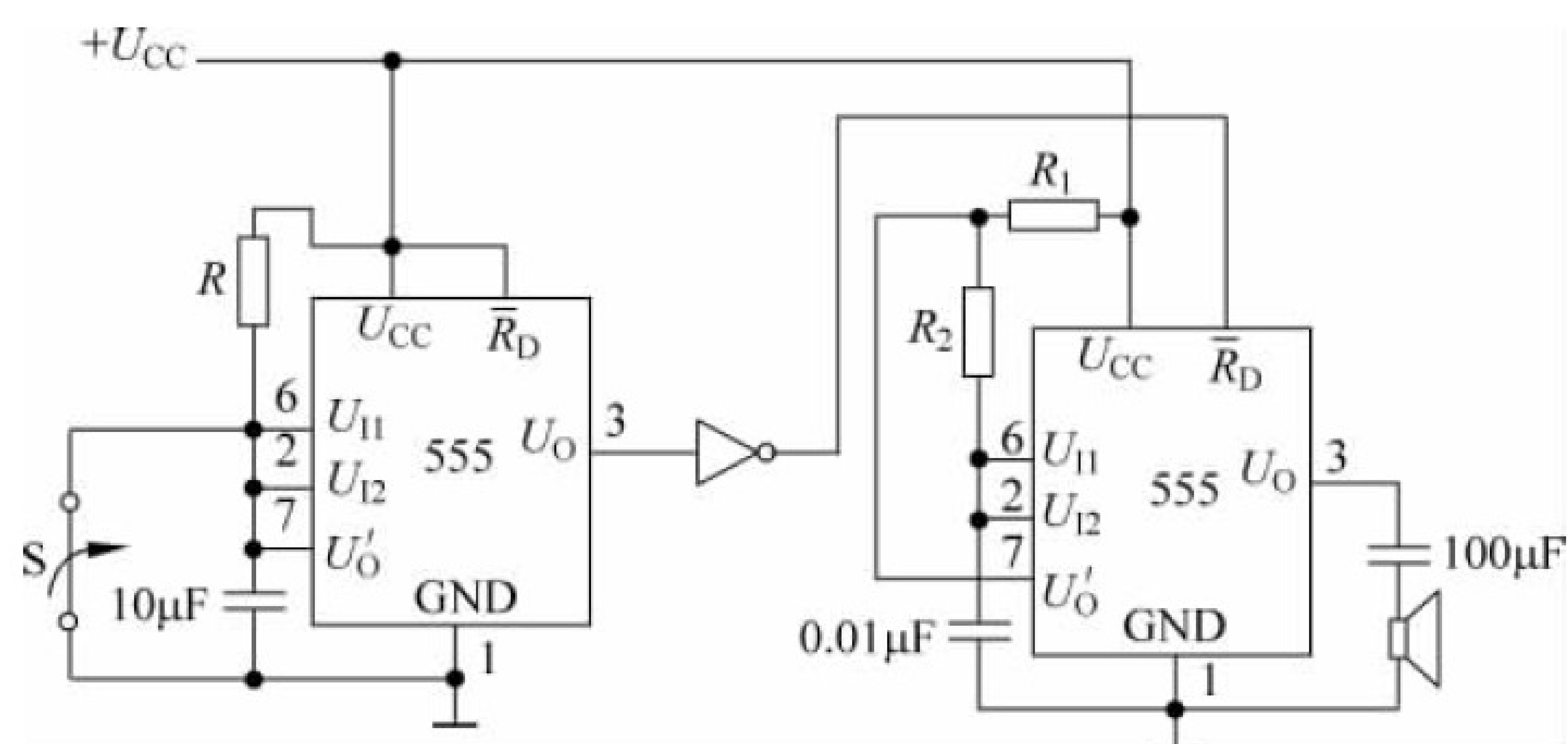


题图 9.9



题图 9.10

23. 如题图 9.11 所示电路中，$R=1\text{M}\Omega$，$R_1=R_2=5\text{k}\Omega$，$U_{CC}=5\text{V}$，求该电路中开关 S 断开后多长时间之内闭合，喇叭不会发声？为什么？

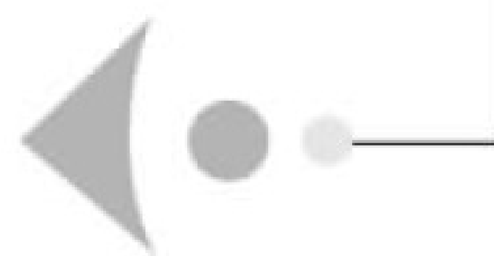


题图　9.11



第10章

利用硬件描述语言描述数字逻辑电路





本章要点：

利用硬件描述语言描述数字逻辑电路是目前数字逻辑电路设计的主流方法之一。学习本章应理解可编程逻辑器件的概念、CPLD 与 FPGA 的编程方法。在此基础上，理解 Verilog HDL 的行为描述方法及门级结构描述方法，初步理解利用 Verilog HDL 描述数字逻辑电路的方法。



10.1 可编程逻辑器件

可编程逻辑器件是 20 世纪 80 年代蓬勃发展起来的数字集成电路，它的出现为数字逻辑电路的设计与实现提供了一条新的思路与途径。

10.1.1 什么是可编程逻辑器件

根据逻辑功能的特点，可将数字集成电路分为通用型和专用型两类。前面各章介绍的中、小规模器件均为通用型数字集成电路。利用这些中、小规模器件，可组成各种复杂的数字系统。随着集成工艺水平的提高，可以将自己设计的数字系统做成一片大规模集成电路，这种为某种专门用途设计的集成电路称为专用集成电路(ASIC)。ASIC 的使用不仅能减小电路的体积、重量、功耗，也能大大提高电路的可靠性。当然，如果用量不大，使用 ASIC 器件将会大幅提高成本，延长设计、生产周期，产生一个很大的矛盾，在很大程度上限制了它的应用。

可编程逻辑器件(Programmable Logic Device，PLD)是一种通用集成电路，它的逻辑功能可按照用户要求通过对器件编程来确定。用户可根据实际应用需要，将 PLD 生产厂家提供的标准结构连接的“与”和“或”(或二者之一)逻辑阵列，按某种规定方式改变 PLD 器件内部的结构，从而获得所需要的逻辑功能。

显然，PROM、EPROM 均是一种可编程逻辑器件。前面介绍的利用 2716 实现逻辑函数的方法事实上便是通过改写存储单元内容改写存储阵列内部的连接从而获得所需要的逻辑功能。当然，EPROM 在大多数场合下均作为存储器使用，人们也更愿意把它当作存储器芯片。

PLD 器件有多种类型，除前面介绍的 PROM、EPROM 外，早期的 PLD 有 PLA(可编程逻辑阵列)、FPLA(现场可编程逻辑阵列)、PAL(可编程阵列逻辑)、GAL(通用阵列逻辑)等简单 PLD(Simple PLD，也称低密度 PLD)。伴随着 PLD 的发展，PLD 器件集成度越来越高，复杂 PLD(Complex PLD，CPLD，也称高密度 PLD)、现场可编程门阵列(Field Programmable Gate Array，FPGA)兴起并得到广泛应用。FPGA 也是 PLD 的一种，只是电路结构上与早期广泛应用的 PLD 差别很大，所以采用 FPGA 这个名字，以示区别。

伴随着 PLD 的发展，其编程设计手段的自动化程度也日益提高。用于 PLD 编程的开发系统由硬件和软件两部分组成。软件部分包括各种编程软件，这些编程软件均有较强的功能，均可运行在普通 PC 上，操作简单方便。新一代的在系统可编程器件(ISP－PLD)的编程就更加简单，只须将计算机运行产生的编程数据直接写入 PLD 即可。例如，前面介绍的 EDA 工具便有在系统可编程的功能。

10.1.2 早期的 PLD 器件

早期 PLD 器件的结构大体相同，主要由与逻辑阵列、或逻辑阵列、输出电路 3 部分组成。

FPLA 器件的与逻辑阵列、或逻辑阵列均可编程，采用的可编程连接元件为熔丝或反熔丝。当可编程连接元件为熔丝时，出厂时是连上的，熔断后为断开状态。当可编程连接元件为反熔丝时，出厂时是断开的，熔合后为连上状态。有兴趣的读者可类比如图 7.3.1 所示的 PROM 存储单元理解 FPLA 器件的编程单元。

PAL 器件的与逻辑阵列可编程，或逻辑阵列为固定形式。早期的 PAL 器件采用的可编程连接元件也为熔丝，后来又出现了采用叠栅 MOS 管为编程器件的 PAL 器件。有兴趣的读者可类比如图 7.3.7 所示的快闪存储器的存储单元理解以叠栅 MOS 管为编程器件的 PAL 器件的编程单元。GAL 器件在 PAL 的基础上改进和发展而来，采用叠栅 MOS 管为编程器件，与逻辑阵列可编程，或逻辑阵列为固定形式。显然，采用熔丝或反熔丝为可编程连接元件的 PLD 器件只可一次编程。采用叠栅 MOS 管为编程器件的 PLD 器件为可多次编程 PLD 器件。

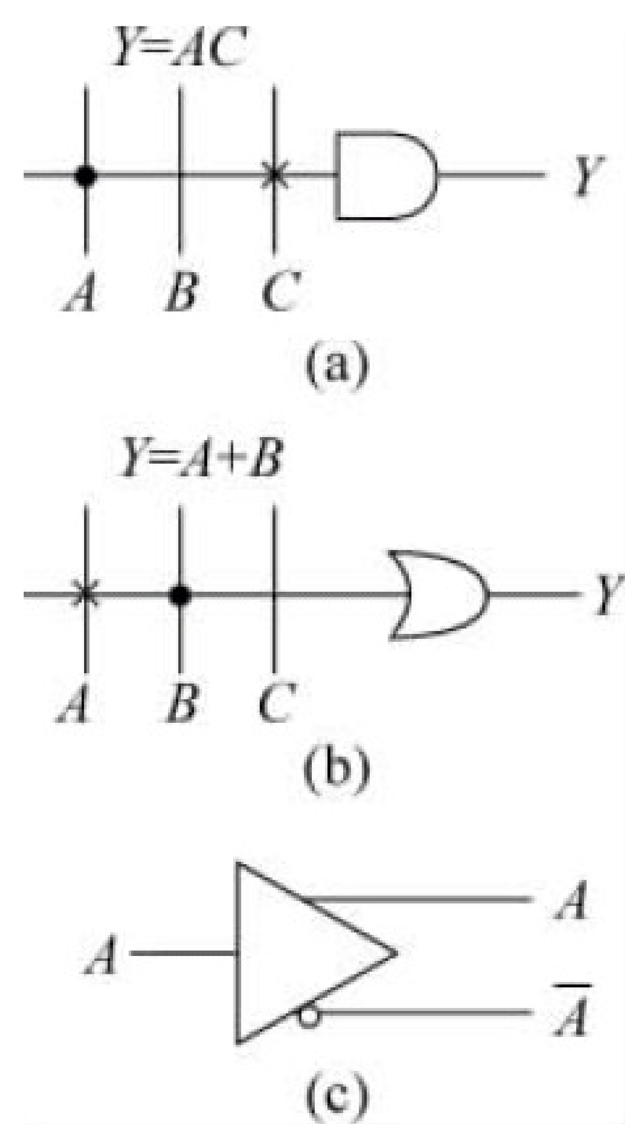


图 10.1.1 PLD 电路中门电路的惯用画法

具体介绍 PLD 的结构前，先来熟悉一下 PLD 电路中门电路的惯用画法。

如图 10.1.1 所示为 PLD 电路中门电路的惯用画法。图中，“·”表示该点为固定连接点，出厂时已做好，用户不能修

改；“×”表示该点为用户编程点，芯片出厂时该点也是接通的，用户可根据需要继续保持接通或将其断开；既无黑点“·”也无叉点“×”表示该点是断开的。所以，图 10.1.1(a)表示 $Y=AC$，图 10.1.1(b)表示 $Y=A+B$，(c)图是互补输出的缓冲器。

可通过如图 10.1.2 所示的 PAL 电路来进一步理解 PLD 器件的结构。

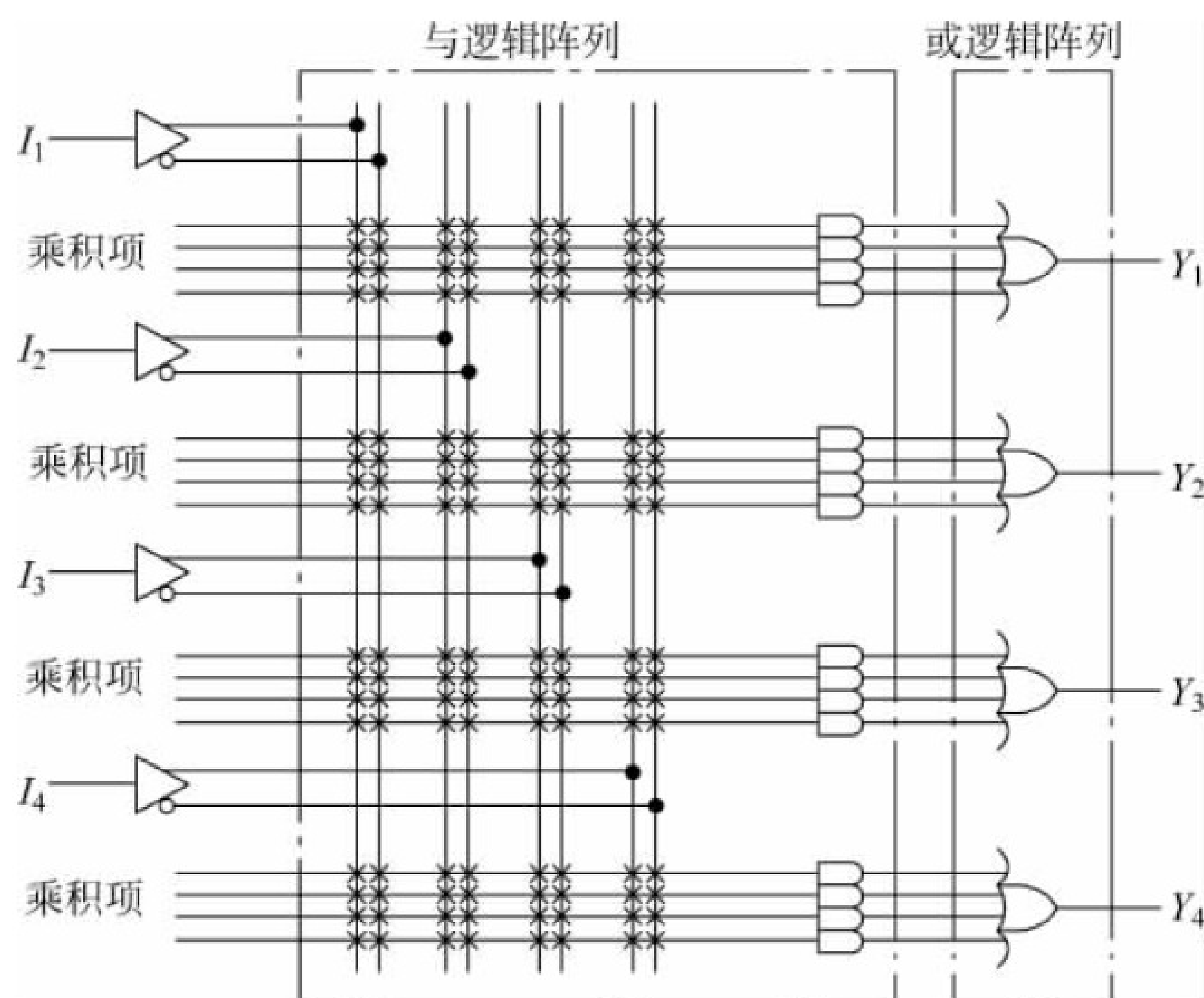


图 10.1.2 未编程的 PAL

如图 10.1.2 所示 PAL 由一个包括 16 个与门的与逻辑阵列、一个由 4 个或门组成的或逻辑阵列和输出缓冲器组成。图中，“·”为固定连接点，出厂时已做好，用户不能修改；“×”表示用户编程点，出厂时该点也是接通的。图中，“与”阵列可编程，“或”阵列固定；编程单元出厂时，所有的交叉点均有熔丝。

比较图 10.1.2 与图 7.3.1，不难发现两图的结构相似(ROM 的存储单元为或逻辑阵列，地址译码器为与逻辑阵列)，也就是说，PAL 与 ROM 结构相似，二者的不同主要有两点：

(1) ROM 的与逻辑阵列是固定的，而 PAL 的与逻辑阵列是可编程的。

(2) ROM 的与逻辑阵列将输入变量最小项全部译出，为全译码工作方式；而 FPLA 的与逻辑阵列是可编程的，可通过编程只产生需要的最小项，为非完全译码方式。

显然，如图 10.1.2 所示的 PAL 是一个未编程的 PLD 芯片，编程后的参考实例如图 10.1.3 所示。

图 10.1.3 中，“×”点为接通的点，由连接关系，有

$$Y_1 = I_1I_2I_3 + I_2I_3I_4 + I_1I_3I_4 + I_1I_2I_4$$
$$Y_2 = \overline{I}_1\overline{I}_2 + \overline{I}_2\overline{I}_3 + \overline{I}_3\overline{I}_4 + \overline{I}_1\overline{I}_4$$
$$Y_3 = I_1\overline{I}_2 + \overline{I}_1I_2$$
$$Y_4 = I_1I_2 + \overline{I}_1\overline{I}_2$$

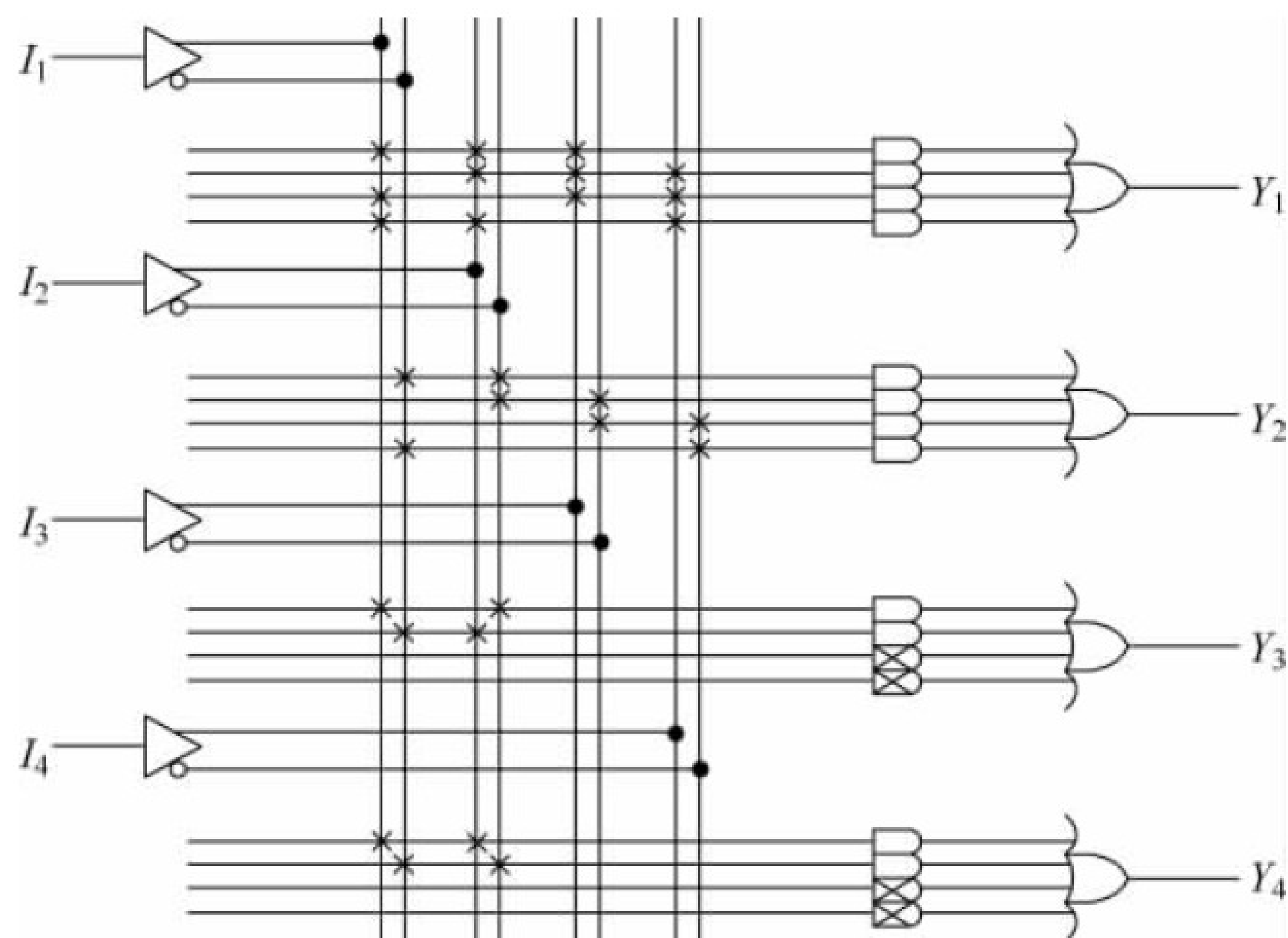


图 10.1.3 编程后的 PAL

如图 10.1.4 所示 FPLA 由一个包括 8 个与门的与逻辑阵列、一个由 4 个或门组成的或逻辑阵列和输出缓冲器组成。图中，“·”为固定连接点，出厂时已做好，用户不能修改；“×”表示用户编程点，出厂时该点也是接通的。FPLA 器件的“与”阵列、“或”阵列均可编程，如图 10.1.4 所示的 FPLA 是一个已编程的 PLD 芯片，当$\overline{\mathrm{OE}}=0$时，有

$$Y_3 = ABCD + \overline{A}\overline{B}\overline{C}\overline{D}$$

$$Y_2 = AC + BD$$

$$Y_1 = A\overline{B} + \overline{A}B = A \oplus B$$

$$Y_0 = CD + \overline{C}\overline{D} = C \odot D$$

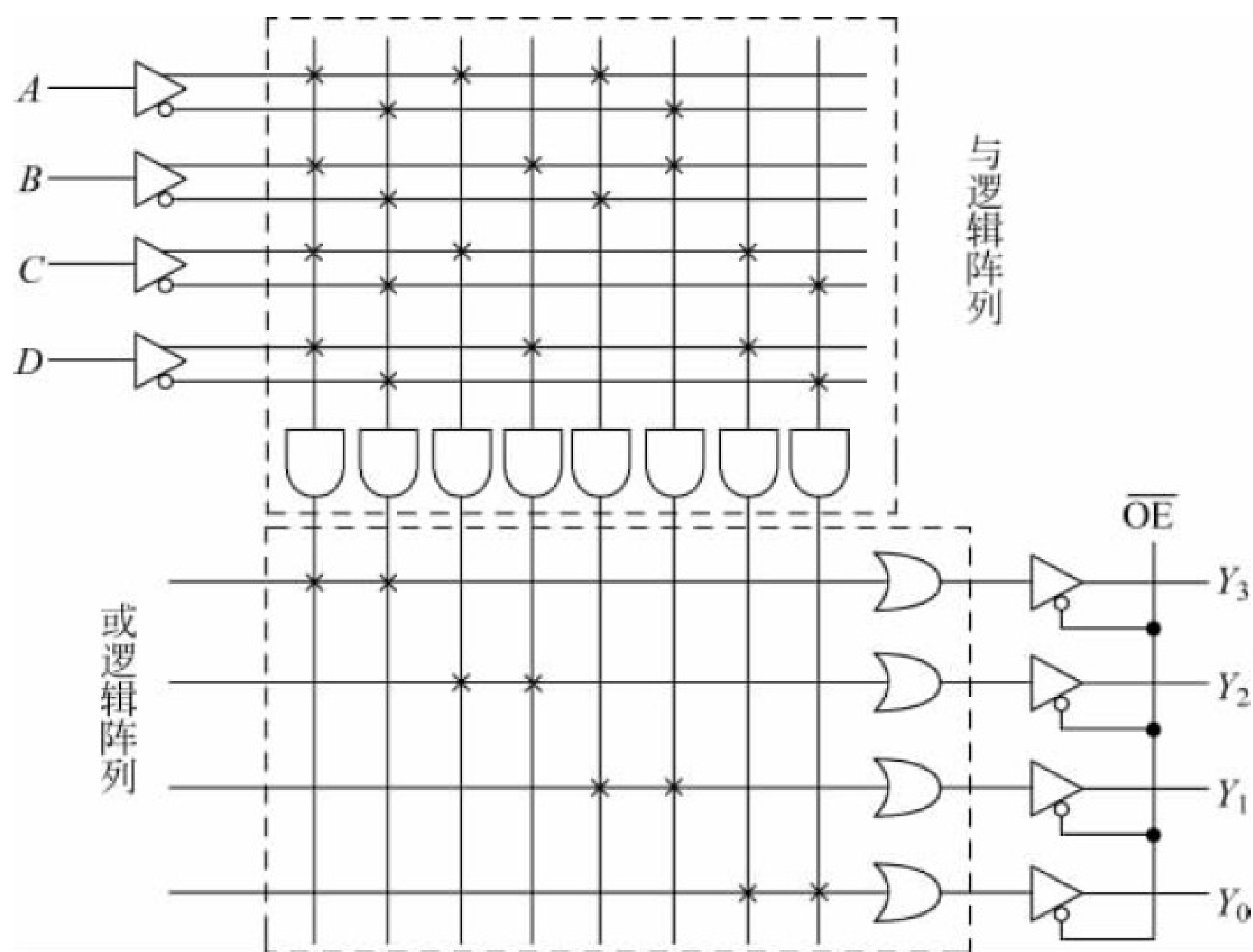


图 10.1.4 编程后的 FPLA

【例 10.1.1】 请用 FPLA 实现下面的函数(例 7.5.3)。

$$Y_1 = A\overline{B}C + \overline{A}\overline{C}D + A\overline{C}, \quad Y_2(A,B,C,D) = \sum_m (0,2,3,4,7,9)$$

解 (1) 分析题意，做出函数的卡诺图。由函数表达式可做出如图 10.1.5 和图 10.1.6 所示的卡诺图。

(2) 求出最简式。由图 10.1.5 和图 10.1.6 所示卡诺图可写出 Y_2、Y_1的最简与或式如下：

$$Y_1 = A\bar{B} + \bar{C}D + A\bar{C}$$

$$Y_2 = \bar{A}\bar{C}\bar{D} + \bar{A}CD + \bar{A}\bar{B}C + A\bar{B}\bar{C}D$$

AB \ CD	00	01	11	10
00		1		
01		1		
11	1	1		
10	**1**	**1**	**1**	**1**

图 10.1.5 Y_1函数的卡诺图

AB \ CD	00	01	11	10
00	1		1	**1**
01	1		1	
11				
10		**1**		

图 10.1.6 Y_2函数的卡诺图

(3) Y_2、Y_1用 FPLA 实现，如图 10.1.7 所示。

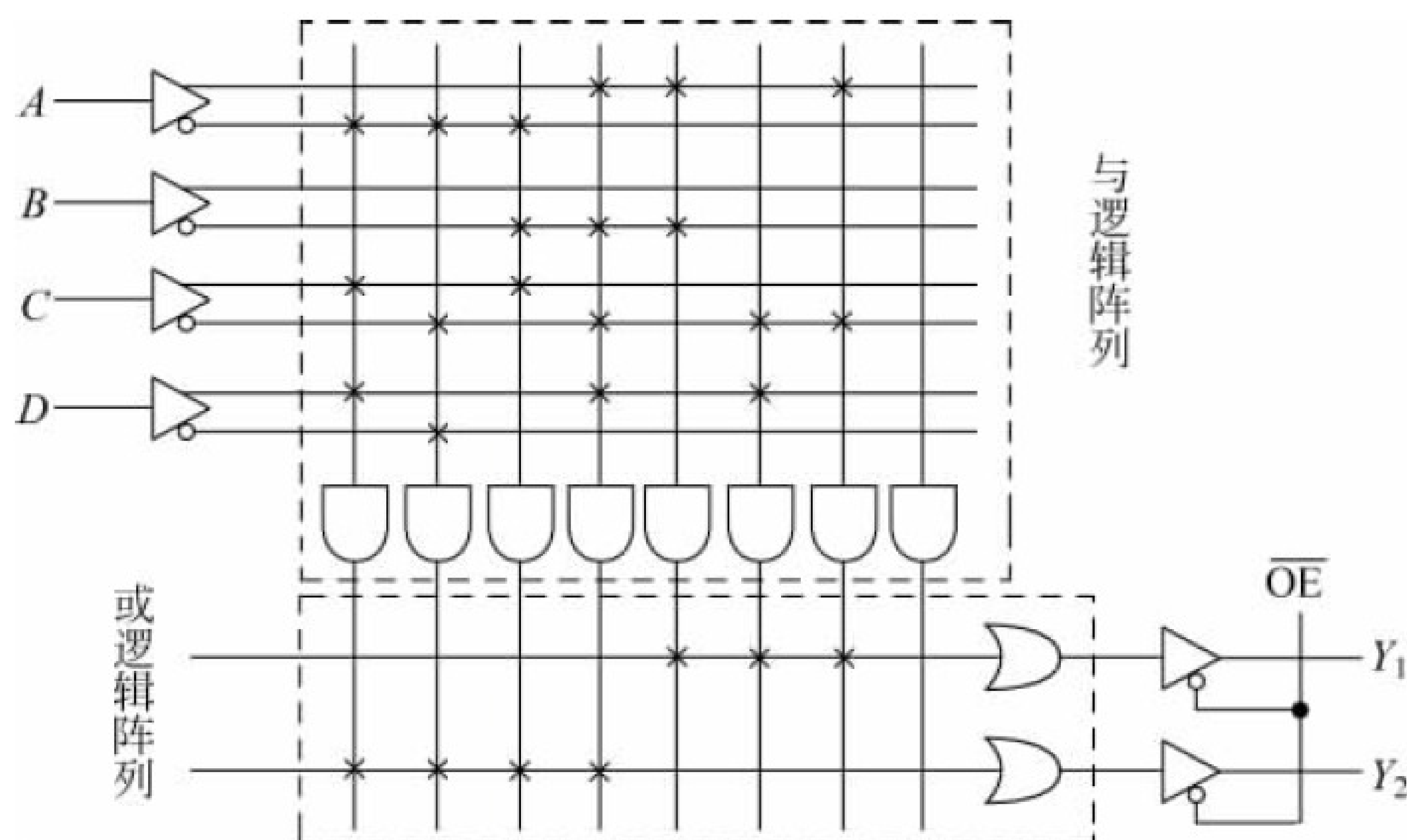


图 10.1.7 FPLA 实现的图

通过例 10.1.1 可以看到，利用 PLD 设计组合电路，编程直观简单，效率高，比用 EPROM 设计组合电路更具优势。当然，早期的 PLD 结构简单，编程当然也简单，但与后期的 PLD 相比，缺乏直观的仿真设计环境，编程手段比较落后。

复杂可编程逻辑器件(Complex Programmable Logic Device，CPLD)是从 PAL 和 GAL 器件发展出来的器件，相对而言规模大、结构复杂。

CPLD 主要由许多逻辑阵列模块(Logic Array Block，LA B)和可编程互连阵列(Programmable Interconnect Array，PLA)组成，参考构成模式如图 10.1.8 所示。每个 LA B 相当于一个 PAL 或 GAL 电路。不同型号的 CPLD 器件可能具有几十个甚至上百个 LA B。通过 PLA 将这些 LAB 连接起来，便构成了各种 CPLD 器件。

有些厂家生产的 CPLD 器件说明书中，把逻辑阵列模块称为功能模块(Function Block)、通用模块(Generic Block)；把可编程互连阵列称为开关矩阵(Switch Matrix)。尽

管不同公司的CPLD器件结构上会有些区别，说法也不尽相同，但构成CPLD的思路是相同的。

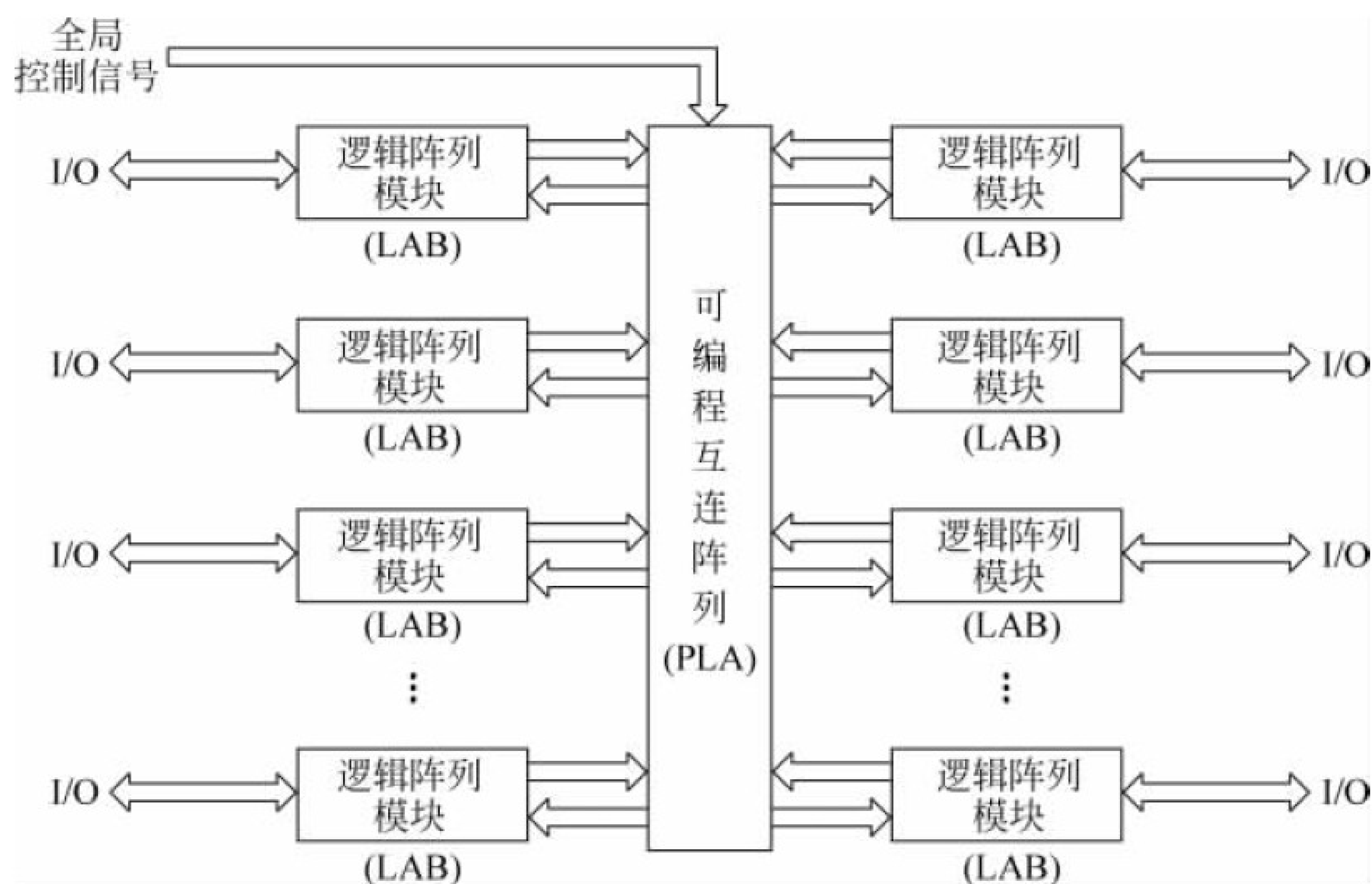


图10.1.8　CPLD的一般结构

CPLD是一种用户可根据个人需要而自行构造逻辑功能的数字集成电路，可借助集成开发软件平台，用原理图、硬件描述语言等方法，生成相应的目标文件，通过下载电缆（“在系统”编程）将代码传送到目标芯片中，实现设计的数字系统。

利用CPLD器件设计数字应用系统的工作大部分是在计算机上完成的。如利用Altera公司的CPLD器件设计数字应用系统时，可打开集成开发软件（Quartus Ⅱ），用画原理图或用硬件描述语言写代码等方法建立要设计系统的模型，编译设计的模型，给出逻辑电路的输入激励信号，进行仿真，查看逻辑输出结果是否正确。设计完成后，进行CPLD芯片引脚输入、输出锁定，完成外围接口电路的设计。之后，生成目标代码，通过下载电缆将代码传送并存储在CPLD芯片后完成整个系统的设计。

在后面的应用实践中，若发现设计的系统有问题，可重新修改原理图或硬件描述语言，完善设计，再次写入代码即可。

现场可编程门阵列（FPGA）是在PAL、GAL、CPLD等可编程器件的基础上进一步发展的产物。PAL、GAL、CPLD等可编程器件均采用了与或逻辑阵列加输出单元的结构形式。FPGA则采用了完全不同的结构形式，由许多“可组态逻辑模块”（Configurable Logic Block，CLB）、输入输出单元（I/O Block，IOB）和分布式的可编程互联矩阵（Programmable Interconnect Matrix，PIM）组成。

FPGA和CPLD都是可编程器件，有很多共同特点。当然，CPLD和FPGA是结构上不同的两种器件，CPLD通过修改具有固定内连电路的逻辑功能来编程，FPGA主要通过改变内部连线的布线来编程；FPGA可在逻辑门下编程，而CPLD是在逻辑块下编程。总体而言，CPLD更适合完成各种算法和组合逻辑应用，FPGA更适合于完成时序逻辑应用。

尽管 FPGA 和 CPLD 结构非常复杂,但使用上确是非常方便的。可利用硬件描述语言描述需要实现的电路模型,调试完成后,生成目标代码,通过下载电缆将代码传送并存储在 FPGA 或 CPLD 芯片后完成整个系统的设计,是目前设计数字应用系统的主流方法之一。在后面的内容中,将主要介绍利用硬件描述语言 Verilog HDL 描述数字逻辑电路的方法。

复习与思考

10.1.1 是否所有 PLD 器件均可多次编程? 为什么?

10.1.2 用 PLD 器件构成的组合逻辑电路和用 2716 存储器实现的组合逻辑电路,哪个效率高,为什么?

10.1.3 试总结 FPGA、CPLD 的编程方法和早期 PLD 编程方法的区别。

10.2 利用 Verilog HDL 描述组合逻辑电路

Verilog HDL 是一种功能强大的硬件描述语言,是一种以文本形式描述数字系统硬件的结构和行为的语言,可用来描述各种逻辑电路。

10.2.1 利用 Verilog HDL 的行为描述方法描述组合逻辑电路



前面设计的各种电路主要以电路结构图的形式作为最终设计成果。硬件描述语言是一种功能强大的高级语言,可不涉及数字电路的结构,通过描述数字电路的行为算法描述数字电路并完成该电路的设计。

行为描述不涉及电路结构,通过行为语句结合算法描述电路的逻辑功能,符合人类逻辑思维方式,可通过下面的例题来理解。

【例 10.2.1】 用 Verilog HDL 的行为描述方法实现一个二输入或门。

解 例 3.5.2 中介绍了二输入或门的描述方法。主要思路为先定义两个输入 a、b,一个输出 y。之后,通过连续赋值 assign 语句利用 C 语言的或逻辑运算符实现。核心语句:

```
assign y = a|b;
```

行为描述不涉及电路结构,可通过直接描述电路的逻辑功能、算法完成电路的设计。基于或运算的逻辑功能,设或门的输入为 a、b,一个输出 y,可用下面的 if 语句描述一个二输入或门,具体如下:

```
if (a == 0 && b == 0) y = 0;
    else y = 1;
```

用 Verilog HDL 的行为描述方法实现的二输入或门完整代码如下:

```
module ort( a,b,y );        //定义模块名、输入、输出
    input a,b;
    output y;
```

```
    reg y;                  //声明 y 为寄存器变量
    always                  //定义 always 块
    begin
        if (a == 0 && b == 0) y = 0;
            else y = 1;
    end
endmodule
```

必须指出，Verilog HDL 是一种高级语言，却又不同于一般的高级语言。Verilog HDL 中的 wire 型(线型)变量并不是普通的高级语言变量，是电路的输入、输出等类型的变量。依照组合逻辑电路的含义，输出(变量)仅与输入(变量)有关，电路中输出发生改变均有特定条件，因此，不可在程序代码中直接修改 wire 型(线型)变量的值，上面的 if 语句不可直接执行，应该改写成 always 块(总是执行)。always 块语法如下：

```
always@(触发条件)
    begin
        多条顺序语句
    end
```

always@(触发条件)含义为给定条件下执行语句块。always 后无@含义为无条件下循环执行语句块。

此外，电路的输出由输入等条件决定，可使用 assign 语句根据输入逻辑关系等给出输出结果。要在代码中修改输出结果，应对输出变量做特殊声明。

```
reg y;  将输出 y 声明为寄存器变量
```

上面代码在 Quartus Ⅱ 中的仿真结果如图 10.2.1 所示。仿真结果显示该代码功能为二输入或门。

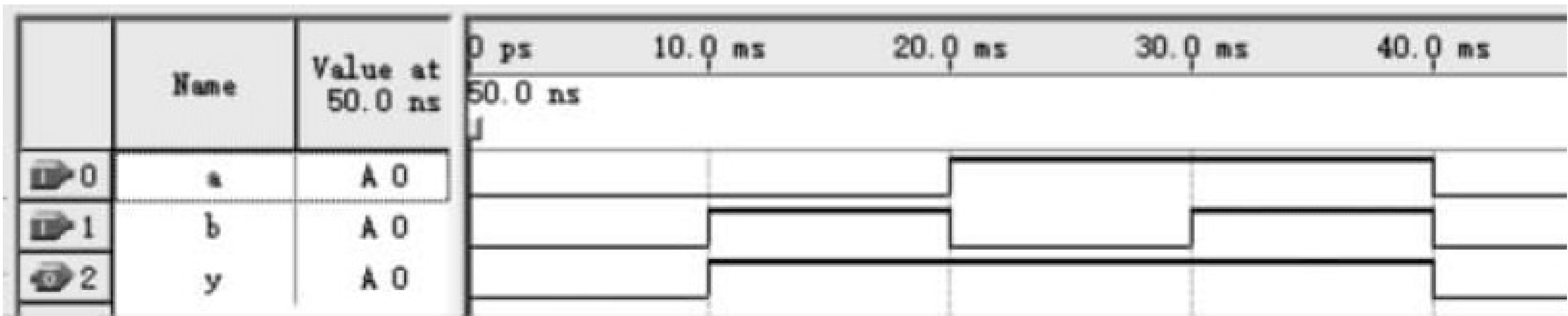


图 10.2.1　Quartus Ⅱ 中仿真结果

【例 10.2.2】 用 Verilog HDL 的行为描述方法实现三变量不一致辨别电路。

解　例 4.2.6 中介绍了利用逻辑代数理论设计实现三变量不一致辨别电路的方法。初步的设计结果用电路实现需要 7 个与非门，之后，通过复杂的变换获得了只需要 5 个与非门实现的表达式。

参考例 10.2.1，直接描述三变量不一致辨别的算法确实非常简单，可用一条复合 if 实现描述，具体如下：

```
if (a == 0 && b == 0 && c == 0 )
                    y = 0;
```

```
        else if (a == 1 && b == 1 && c == 1)
                y = 0;
            else   y = 1;
```

三变量一致有两种情况：输出为0；输出为1。实现三变量不一致辨别电路的完整代码如下：

```
module ort( a,b,c,y );       //定义模块名、输入、输出
   input a,b,c;
   output y; reg y;          //声明 y 为寄存器变量
always                       //定义 always 块
begin
    if (a == 0 && b == 0 && c == 0 )            y = 0;
        else if (a == 1 && b == 1 && c == 1)    y = 0;
            else   y = 1;
     end
endmodule
```

上面代码在 Quartus Ⅱ 中的仿真结果如图 10.2.2 所示。仿真结果显示该代码功能为三变量不一致辨别电路。

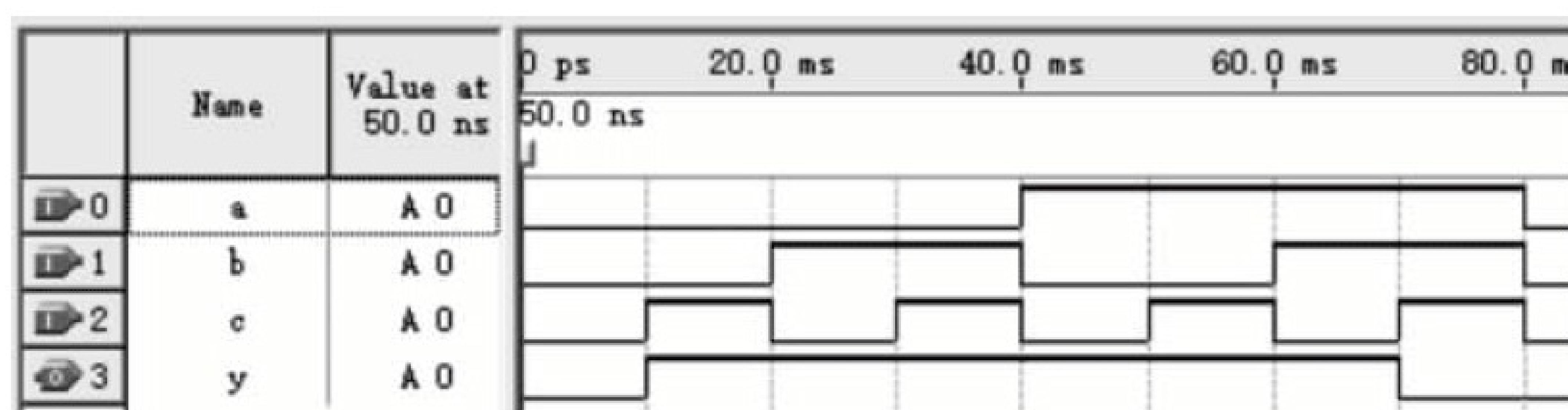


图 10.2.2 Quartus Ⅱ 中仿真结果

【例 10.2.3】 某医院有1、2、3、4号病房4间，装有4个呼叫器，对应的护士室有1、2、3、4号4个指示灯。优先级按照1、2、3、4顺序降低设置，请用 Verilog HDL 的行为描述方法实现该控制电路。

解 设有4个输入 a、b、c、d，对应1、2、3、4号病房的4个呼叫器，呼叫器按钮按下时输出低电平。设有4个输出 L1、L2、L3、L4，对应的护士室有1、2、3、4号4个指示灯，输出为1时灯亮。

依照功能要求，实现该控制电路的算法如下：

```
L1 = 0;L2 = 0;L3 = 0;L4 = 0;
if (a == 0 ) L1 = 1;
    else if (b == 0) L2 = 1;
        else if (c == 0) L3 = 1;
            else if (d == 0) L4 = 1;
```

默认情况下，4个灯均不亮，L1＝L2＝L3＝L4＝0；当有按钮按下时，优先级高的按钮对应的灯亮。该控制电路的完整代码如下：

```
module ort(a,b,c,d,L1,L2,L3,L4);
    input a,b,c,d;
    output L1,L2,L3,L4;
    reg L1,L2,L3,L4;
    always
        begin
          L1 = 0;L2 = 0;L3 = 0;L4 = 0;
          if (a == 0) L1 = 1;
              else if (b == 0) L2 = 1;
                  else if (c == 0) L3 = 1;
          else if (d == 0) L4 = 1;
    end
endmodule
```

上面代码在 Quartus Ⅱ中的仿真结果如图 10.2.3 所示。仿真结果显示该代码实现的控制逻辑正确。

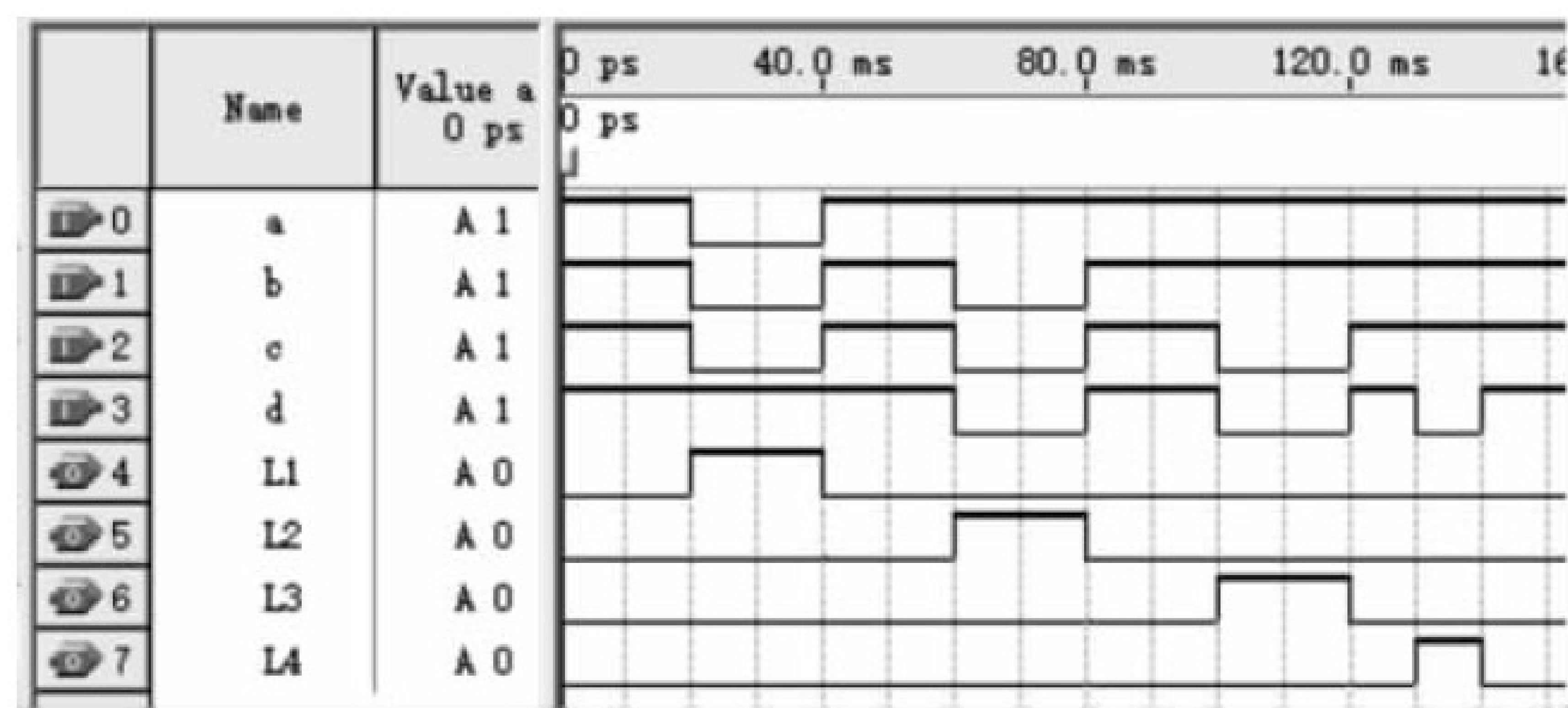


图 10.2.3 Quartus Ⅱ中仿真结果

10.2.2 利用 Verilog HDL 的门级结构描述方法描述组合逻辑电路

当然，电路的功能最终是由其结构确定的，通过描述数字系统的行为算法实现的数字系统最终也是通过逻辑门的连接关系实现的。硬件描述语言当然可通过描述逻辑门以及逻辑门之间的连接关系设计实现数字逻辑电路。



可通过下面的例题来理解。

【例 10.2.4】 用 Verilog HDL 的门级结构描述方法描述如图 10.2.4 所示电路。

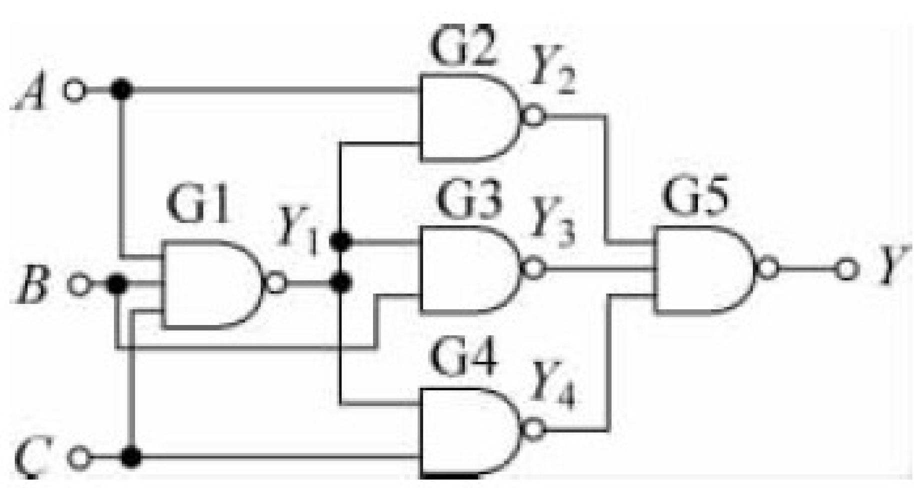


图 10.2.4 例 10.2.4 的图

解 具体代码如下：

```
module ort(A,B,C,Y);
   input A,B,C;
   output Y;
   wire Y1,Y2,Y3,Y4;          //声明 4 个中间变量
   nand G1(Y1,A,B,C);         //描述与非门 G1
   nand G2(Y2,A,Y1);
   nand G3(Y3,B,Y1);
   nand G4(Y4,C,Y1);
   nand G5(Y,Y2,Y3,Y4);
```

```
endmodule
```

如图 10.2.4 所示电路具有 3 个输入(A,B,C)、1 个输出 Y,描述代码如前 3 行所示。

如图 10.2.4 所示电路还具有 4 个中间输出变量 Y1、Y2、Y3、Y4,应声明 4 个中间变量,描述代码如第 4 行所示。

输入、输出变量及中间变量定义好后,可具体地描述各门电路的连接关系,描述代码如第 5~9 行所示。如第 5 行的"nand G1(Y1,A,B,C);",nand 声明后面的门电路类型为与非门,G1 为门的序号,()中的变量为该的输出变量 Y1 及 3 个输入变量(A,B,C)。可类似地分析第 6~8 行代码的含义。

当然,门的序号是门的标识,应唯一。第 5 行的门序号为 G1,后面各门不可再定义为 G1。声明门电路类型的主要关键词有 and(与)、or(或)、not(非)、nand(与非)、nor(或非)、xor(异或)等。

上面代码在 Quartus Ⅱ 中仿真结果如图 10.2.2 所示。仿真结果显示该代码功能为三变量不一致辨别电路。三变量一致的两种情况下输出为 0,其余情况下输出为 1。由例 4.2.2 可知,如图 10.2.4 所示电路为三变量不一致辨别的电路。

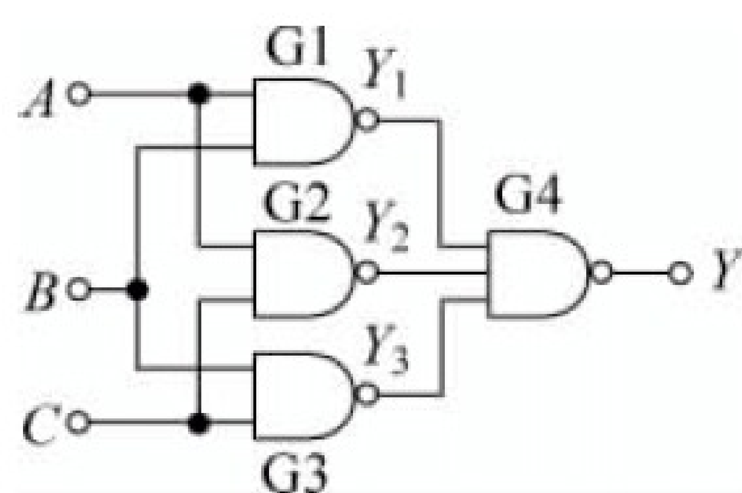


图 10.2.5　例 10.2.5 的图

【例 10.2.5】 用 Verilog HDL 的门级结构描述方法描述如图 10.2.5 所示电路。

解　具体代码如下:

```
module ort(A,B,C,Y);
   input A,B,C;
   output Y;
   wire Y1,Y2,Y3;
   nand G1(Y1,A,B);
   nand G2(Y2,A,C);
   nand G3(Y3,B,C);
   nand G4(Y,Y1,Y2,Y3);
endmodule
```

如图 10.2.5 所示电路具有 3 个输入(A,B,C)、1 个输出 Y,描述代码如前 3 行所示。

如图 10.2.4 所示电路还具有 3 个中间输出变量 Y1、Y2、Y3,应声明 3 个中间变量,描述代码如第 4 行所示。

输入、输出变量及中间变量定义好后,可具体地描述各门电路的连接关系,描述代码如第 5~8 行所示。

上面代码在 Quartus Ⅱ 中仿真结果如图 4.2.6 所示。仿真结果显示该代码功能为 3 人表决逻辑电路。即 3 个人对一个提案进行表决,若赞成则该提案用"1"表示,否则用"0"表示。如果有 2 个或者 2 个以上的人赞成该提案,则该提案就被通过,电路 Y 输出"1";否则,该提案被否决。由例 4.2.3 可知,如图 10.2.5 所示电路为 3 人表决逻辑电路。

【例 10.2.6】 用 Verilog HDL 的门级结构描述方法描述如图 10.2.6 所示的 2 位串行

进位加法器。

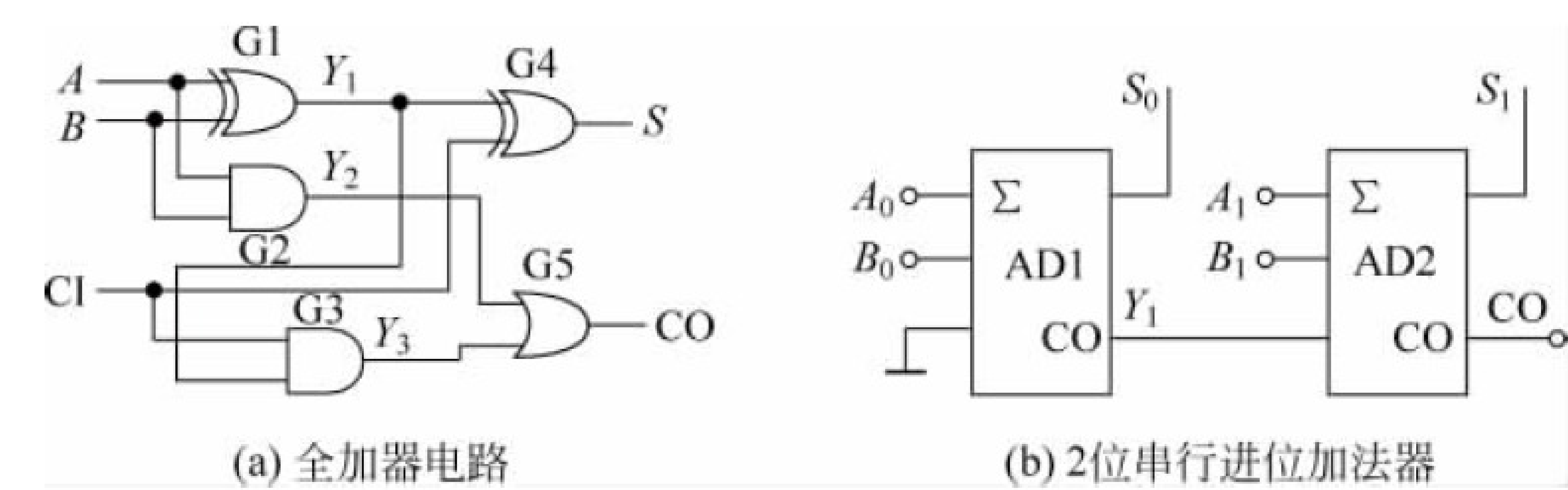


图 10.2.6　例 10.2.6 的图

解　当数字系统比较复杂时，一般采用自上而下或自下而上的方法描述数字系统。本题采用自下而上的方法描述 2 位串行进位加法器。2 位串行进位加法器由全加器构成，应先描述底层元件全加器的电路结构。可用 1 个底层文件“addf. V”描述如图 10.2.6(a)所示全加器，具体代码如下：

```
module addf(A,B,CI,S,CO); //模块名应与文件名一致
   input A,B,CI;
   output CO,S;
   wire Y1,Y2,Y3;
   xor G1(Y1,A,B);
   and G2(Y2,A,B);
   and G3(Y3,Y1,CI);
   xor G4(S,CI,Y1);
   or G5(CO,Y2,Y3);
endmodule
```

在上面的代码中，xor 为系统定义的异或门，or 为系统定义的或门，and 为系统定义的与门。仿真结果如图 10.2.7 所示。仿真结果显示该代码功能为全加器。A、B 为两个一位的二进制数；CI 为来自低位的进位数；S 为相加后的和数；CO 为向高一位的进位。

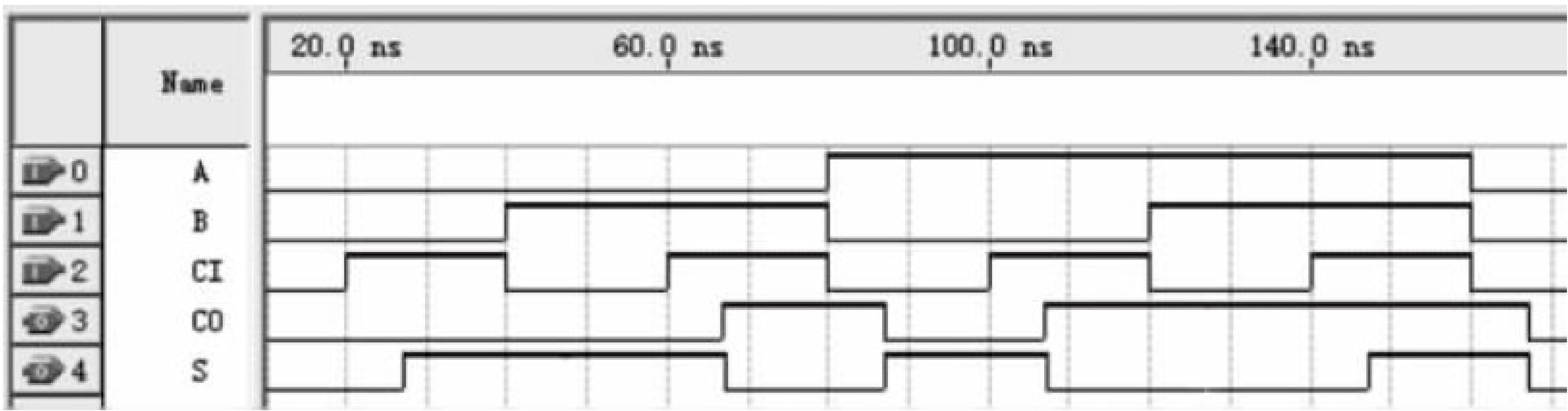


图 10.2.7　全加器的仿真图

上面的代码描述了全加器，要求模块名与文件名相同。该代码对应文件名为 addf. V，模块名称为“addf”。可利用这个定义好的模块定义具体的实例，可用一个顶层文件“ort. V”描述 2 位串行进位加法器，顶层文件代码如下：

```
module ort(A,B,CI,S,CO); //模块名应与文件名一致
  input [1:0] A,B;
  input CI;
```

```
  output CO;
  output [1:0] S;
  wire Y1;
  addf AD1(A[0],B[0],CI,S[0],Y1); //描述第 1 个全加器
  addf AD2(A[1],B[1],Y1,S[1],CO);
endmodule
```

在上面的代码中,“input [1:0] A,B;”定义了一组输入 A,B,4 个输入包括 A[0]、A[1]、B[0]、B[1] 。类似地,“output [1:0] S;”定义了一组输出 S,包括 2 个输出 S[0]、S[1]。

在上面的代码中,addf 为底层文件 addf.V 定义的全加器模块。依照如图 10.2.6(b)所示 2 位全加器的结构,可利用 addf 创建了两个全加器的实例。如“addf AD1(A[0],B[0],CI,S[0],Y1);”,创建了一个名为 AD1 的全加器实例,实例中的输入、输出与 addf.V 中定义的全加器模块的输入、输出一一对应。可见,全加器 AD1 完成了 A[0],B[0] 两个 1 位二进制数的全加运算,进位 Y1 作为中间变量送往全加器 AD2。

顶层文件“ort.V”的仿真结果如图 10.2.8 所示。仿真结果显示该代码功能为两个 2 位二进制数的加法运算电路。A、B 为两个 2 位的二进制数;CI 为来自低位的进位数;S 为相加后的和数;CO 为向高一位的进位。

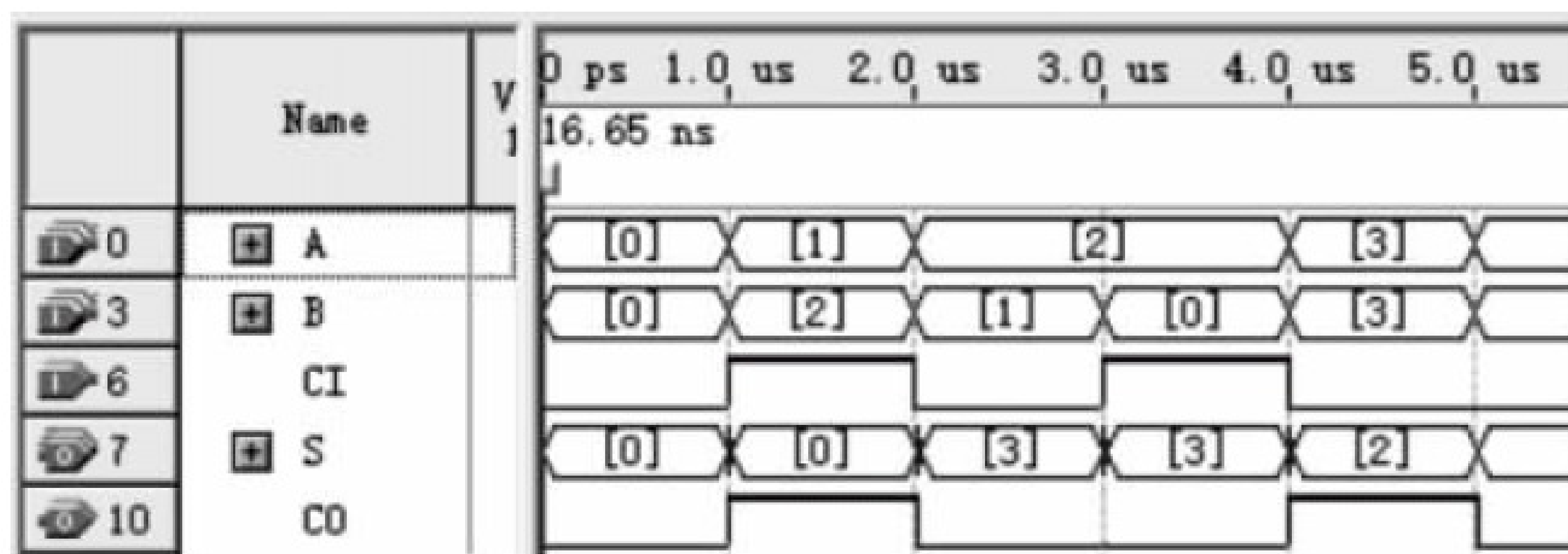


图 10.2.8　最终的仿真结果

复习与思考

10.2.1　例 10.2.3 的描述代码中,去掉“L1=0;L2=0;L3=0;L4=0;”,是否还能实现要求的功能?

10.2.2　例 10.2.5 中,“nand G1(Y1,A,B);”“nand G2(Y2,A,C);”两行交换顺序对电路的逻辑功能有无影响?

10.2.3　试总结 Verilog HDL 的行为算法描述方法和门级结构描述方法的各自优缺点。

10.3　利用 Verilog HDL 描述时序逻辑电路

利用 Verilog HDL 的行为算法描述方法或门级结构描述方法描述时序逻辑电路是目前设计时序逻辑电路的主流方法之一。

10.3.1 利用 Verilog HDL 描述触发器

触发器是构成时序逻辑电路的基础，下面通过几个实例来了解利用 Verilog HDL 描述触发器的方法。

【例 10.3.1】 用 Verilog HDL 描述一个支持异步清零上升沿触发的 D 触发器。

解 描述代码如下：

```
module ddf(d,cp,rd,q);              //定义模块名、输入、输出
   input d,cp,rd;
   output q;
   reg q;                           //声明输出状态 q 为寄存器变量
always @(posedge cp or negedge rd)
    // always 块的触发条件为 cp 上升沿或 rd 下降沿
     begin
      if(!rd) q<=1'b0;              //q=0
        else q<=d;                  //q=d
     end
endmodule
```

在上面的代码中，采用行为描述方法描述 D 触发器。定义了时钟 cp、外部输入 d、异步清零 rd 3 个输入及输出 q，将 q 声明为寄存器变量，方便 always 块修改其值。根据题目要求，always 块的触发条件为 cp 上升沿，描述代码为"posedge cp"。题中还要求具有异步清零功能。设异步清零 rd 为低电平有效，正常 rd 为 1，因此，always 块还有一个触发条件为 rd 下降沿，描述代码为"negedge rd"。

Verilog HDL 中的整数表示方法为：[size]'base value。其中，size 是可选项，描述了整数的长度。base 表示这个数据的进制，包括 b(或 B，二进制)、o(或 O，八进制)、d(或 D，十进制)、h(或 H，十六进制)等 4 种进制。1' b0 的含义为一个二进制位，值为 0。

always 块中变量赋值有两种方法：阻塞赋值"="；非阻塞赋值"<="。当使用"="赋值时，给一个变量赋值的同时，阻塞另一个变量赋值，块中各变量先后赋值。当使用"<="赋值时，所有变量被同时赋值。

代码"q<=1' b0"的含义为采用非阻塞赋值方法将 0 赋给 q。显然，当有多条这样的赋值语句时，本次赋值不影响后面各变量的赋值。如有"q<=1' b0"；"q1<=q；"两条赋值语句时，q=0，q1=q^n，非常吻合同步时序电路的状态变化规律，因此，描述时序逻辑电路时，常使用"<="赋值。

上面代码的仿真结果如图 10.3.1 所示。仿真结果显示该代码功能为支持异步清零上升沿触发的 D 触发器。当 rd=0 时，触发器立即清零；当 rd=1，cp 上升沿到来时，q=d。

【例 10.3.2】 请用 Verilog HDL 描述一个支持异步清零上升沿触发的 JK 触发器。

解 描述代码如下：

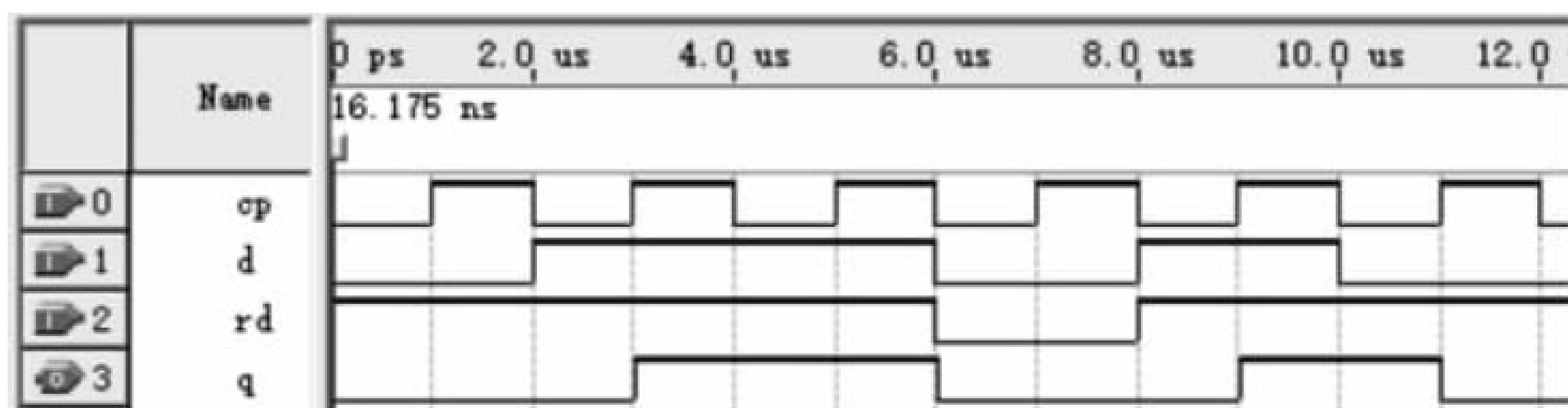


图 10.3.1 例 10.3.1 的图

```
module jksff(j,k,cp,rd,q);
   input j,k,cp,rd;
   output q;
   reg q;
always @(posedge cp or negedge rd)
      begin
        if(!rd) q<=1'b0; //清零
          else q<=(j&(~q))|((~k)&q);
      end
endmodule
```

在上面的代码中，采用行为描述方法描述 JK 触发器。定义了时钟 cp，外部输入 J、K，异步清零 rd 4 个输入及输出 q，将 q 声明为寄存器变量，方便 always 块修改其值。

JK 触发器的特征方程为

$$Q^{n+1}=J\overline{Q}^{n}+\overline{K}Q^{n}$$

写成 Verilog 代码为

```
q<=(j&(~q))|((~k)&q)
```

上面代码的仿真结果如图 10.3.2 所示。仿真结果显示该代码功能为支持异步清零上升沿触发的 JK 触发器。当 rd=0 时，触发器立即清零；当 rd=1，cp 上升沿到来时，$Q^{n+1}=J\overline{Q}^{n}+\overline{K}Q^{n}$。

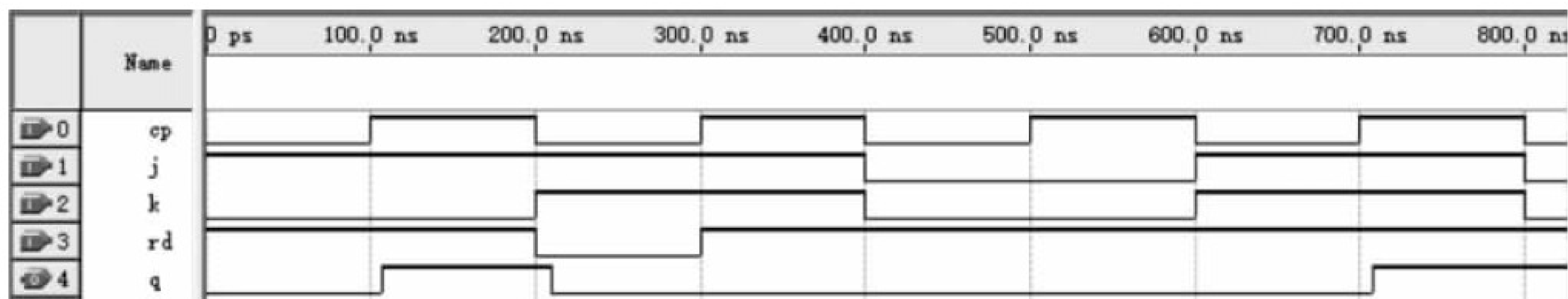


图 10.3.2 例 10.3.2 的图

10.3.2 利用 Verilog HDL 描述时序逻辑电路

【例 10.3.3】 请用 Verilog HDL 描述如图 10.3.3 所示电路，图中各触发器均为上升沿触发的 JK 触发器。

解 如图 10.3.3 所示电路包括 3 个 JK 触发器，采用自下而上的方法进行描述。应先

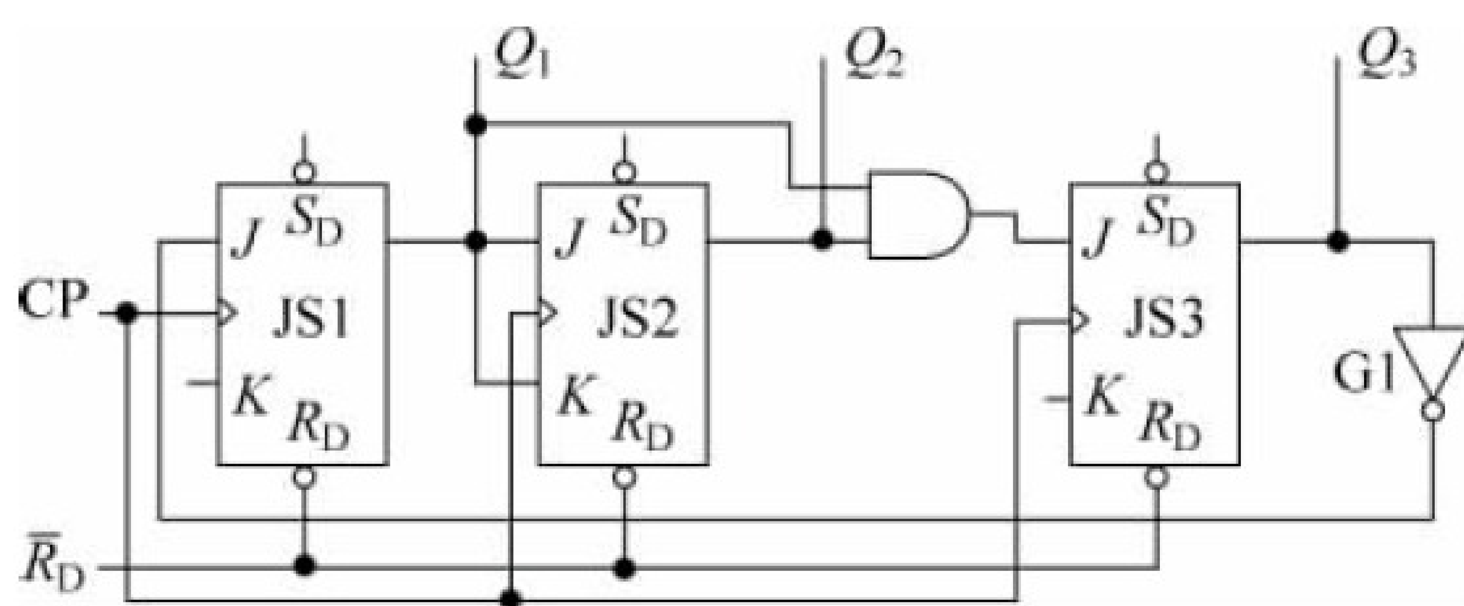


图 10.3.3　例 10.3.3 的图

描述底层元件上升沿触发的 JK 触发器的电路结构，描述方法如例 10.3.2 所示，底层文件名“jksff. V”，模块名“jksff”。

可用一个顶层文件“five. V”描述如图 10.3.3 所示电路，顶层文件代码如下：

```
module five(CP,ST,Q);
  input CP,ST;
  output [2:0] Q;
  wire Y1;
  jksff JS1(Y1,1,CP,ST,Q[0]);
  jksff JS2(Q[0],Q[0], CP,ST,Q[1]);
  jksff JS3(Q[0]&Q[1], 1, CP,ST,Q[2]);
  not G1(Y1,Q[2]);
endmodule
```

在上面的代码中，“input CP，ST；”定义了外部时钟 CP 和异步清零 ST 2 个输入；“output [2:0] Q”定义了一组输出 Q，包括 Q[0]、Q[1]、Q[2] 3 个输出。

在上面的代码中，jksff 为底层文件 jksff. V 定义的 JK 触发器，可利用 jksff 创建 3 个 JK 触发器。如“jksff JS1(Y1，1，CP，ST，Q[0])；”，创建了一个名为 JS1 的 JK 触发器实例，输入 J＝Y1（反相器 G1 的输出）、K＝1，时钟接外部时钟 cp，清零端接外部清零输入 ST，输出为 Q[0]。可类似分析 JS2、JS3。“not G1(Y1，Q[2])；”描述了反相器 G1，输出 Y1 作为中间变量送往 JS1。

顶层文件“five. V”的仿真结果如图 6.2.4 所示。图中，$Q_3Q_2Q_1$ 从状态 0 开始，经 5 个时钟，历 4 个状态后回到 0，为对时钟信号计数的同步五进制加法计数器，与例 6.2.1 的分析结果吻合。

【例 10.3.4】 请用 Verilog HDL 描述一个一百进制计数器。

解 一百进制计数器具有 100 个计数状态，可用 8 个二进制位表示。参照 74LS161 的引脚图，可定义 out 为计数器的输出，data 为计数器的预置数，load、reset、en 分别对应置数、清零、计数允许等控制引脚，co 为进位输出，clk 为时钟。

设置数、清零均为低电平有效，采用同步方式；计数允许为高电平有效，满幅时进位输出有效，为高电平。可写出描述代码如下：

```
module count100(out,data,load,reset,en,clk,co);
   output [7:0] out;                    //计数器的输出
```

```
    input [7:0] data;                    //计数器的预置数
    input load,reset,clk,en;             // 置数、清零、计数允许、时钟
    output co;                           //进位信号
    reg [7:0] out;                       //声明为寄存器变量
    reg co;
always @(posedge clk)                    //上升沿触发
      begin
       if(!reset)out = 8'd000;           //清零
         else if (!load) out = data;     //置数
           else if (en) out = out + 1;   //加法计数
      if (out >= 100) out = 8'd000;      //溢出归零
      if (out == 99) co = 1;             //满幅进位有效
             else co = 0;
      end
endmodule
```

上面代码的仿真结果如图 10.3.4 所示。图中，$Q_7 \sim Q_0$ 从状态 0 开始加法计数，进入状态 63_{16} 后 CO=1，当下一个时钟到来时回到 0，为对时钟信号计数的同步一百进制加法计数器。

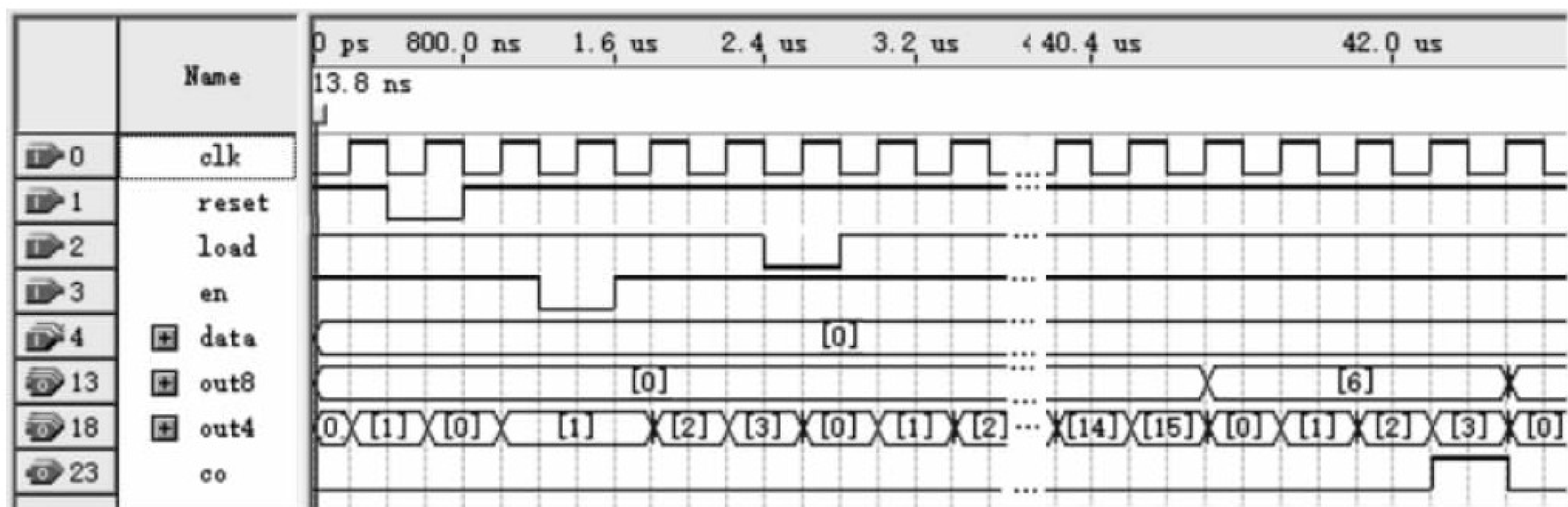


图 10.3.4　例 10.3.4 的仿真结果

复习与思考

10.3.1　总结阻塞赋值"＝"和非阻塞赋值"＜＝"二者的异同？

10.3.2　参考例 10.3.4 设计一个具有异步清零、同步置数功能的一百进制计数器。

习题

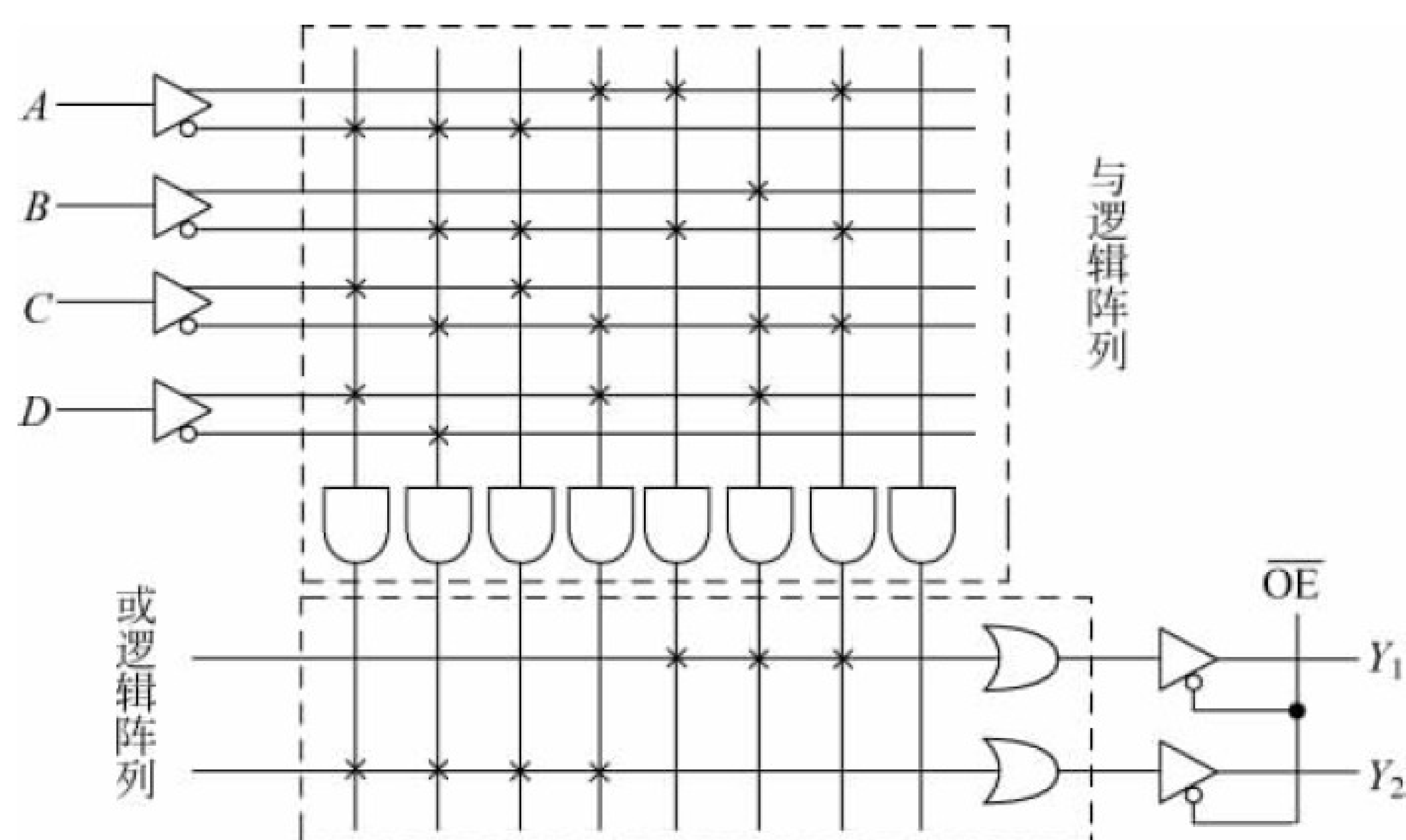
10-1　填空题

1. ________简称 PLD，是一种通用集成电路，它的逻辑功能可按照用户要求通过________来确定。EPROM 中的数据可以修改，因此，EPROM 也是一种________，只是在大多数场合下，人们更愿意把它当作存储器芯片使用。

2. always 块中变量赋值有________"＝"、________"＜＝"两种方法。赋值时，给一个变量赋值的同时，阻塞另一个变量赋值，这种赋值方式为________。描述时序逻辑电路时，常使用"________"赋值。

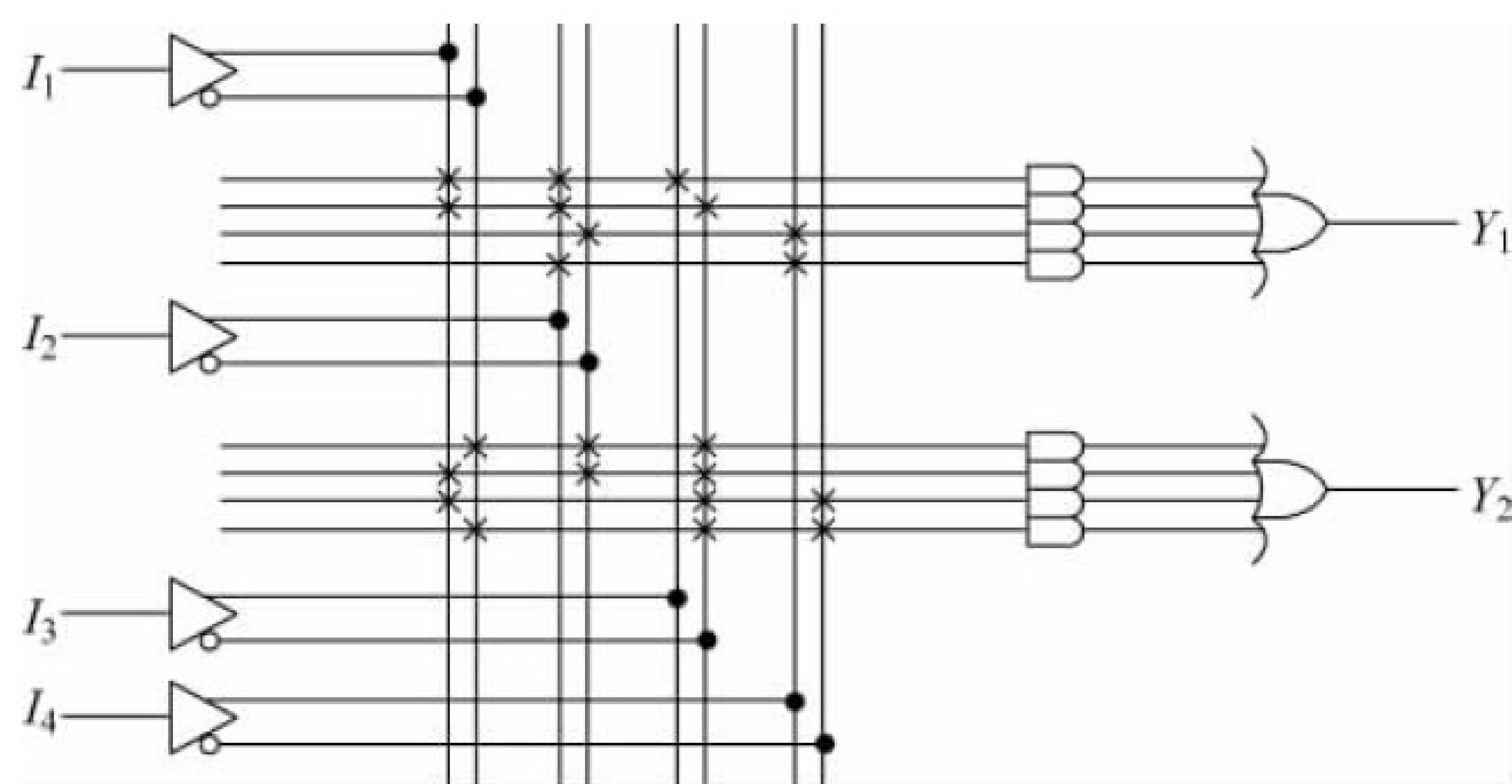
10-2　分析应用题

1. 试写出如题图 10.1 所示 FPLA 的逻辑函数。



题图　10.1

2. 试写出如题图 10.2 所示 PAL 的逻辑函数。



题图　10.2

3. 用 FPLA 实现下面的函数。

(1) $Y_1 = A\overline{B}CD + \overline{A}B\overline{C} + A\overline{C}D$

(2) $Y_2(A,B,C,D) = \sum_m (0,1,2,3,4,8,9,10,11)$

4. 分析下面代码的逻辑功能。

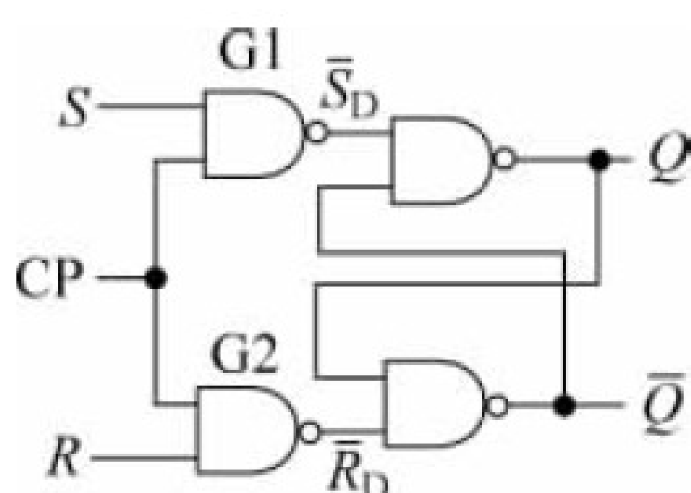
```
module xort( a,b,y );
   input a,b;
   output y;
   reg y;
   always
    begin
       if (a == b) y = 0;
            else y = 1;
    end
endmodule
```

5. 分析下面代码对应的电路图。

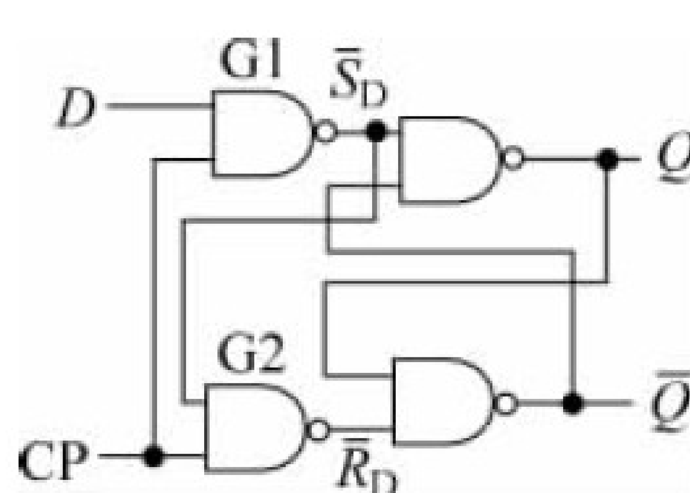
```
module ort(A,B,C,Y);
   input A,B,C;
   output Y;
   wire Y1,Y2,Y3;
   nand G1(Y1,A,B,C);
   nand G2(Y2,A,Y1);
   nand G3(Y3,B,Y1);
   nor G4(Y,Y2,Y3);
endmodule
```

6. 用 Verilog HDL 的门级结构描述方法描述如题图 10.3 所示的电路。

7. 用 Verilog HDL 的门级结构描述方法描述如题图 10.4 所示的电路。

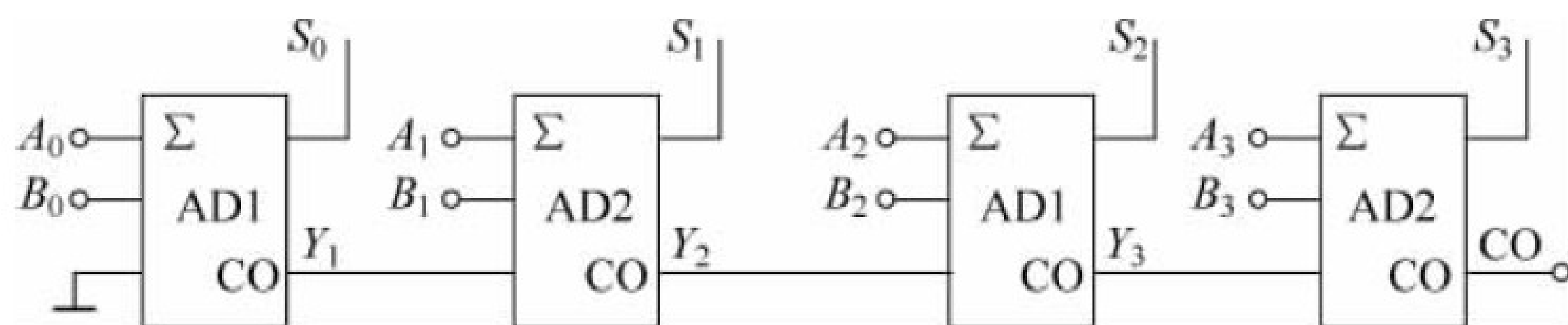


题图 10.3



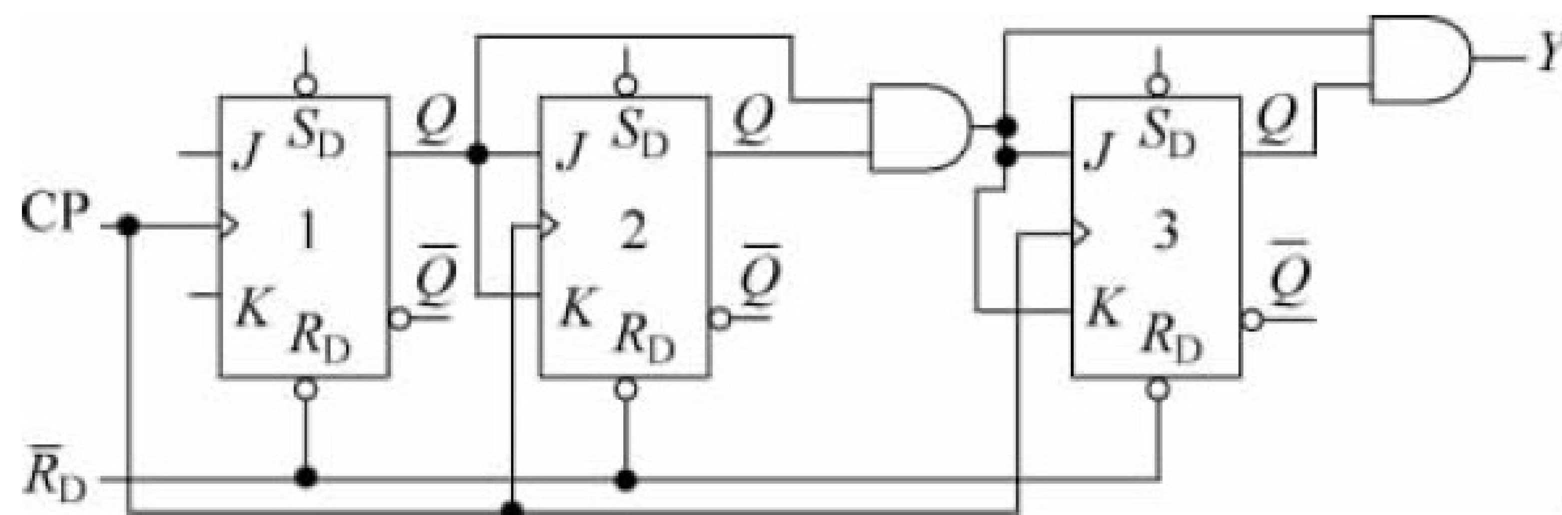
题图 10.4

8. 用 Verilog HDL 的门级结构描述方法描述如题图 10.5 所示的电路，图中的 1 位全加器模块如例 10.2.6 所示，已经定义好。



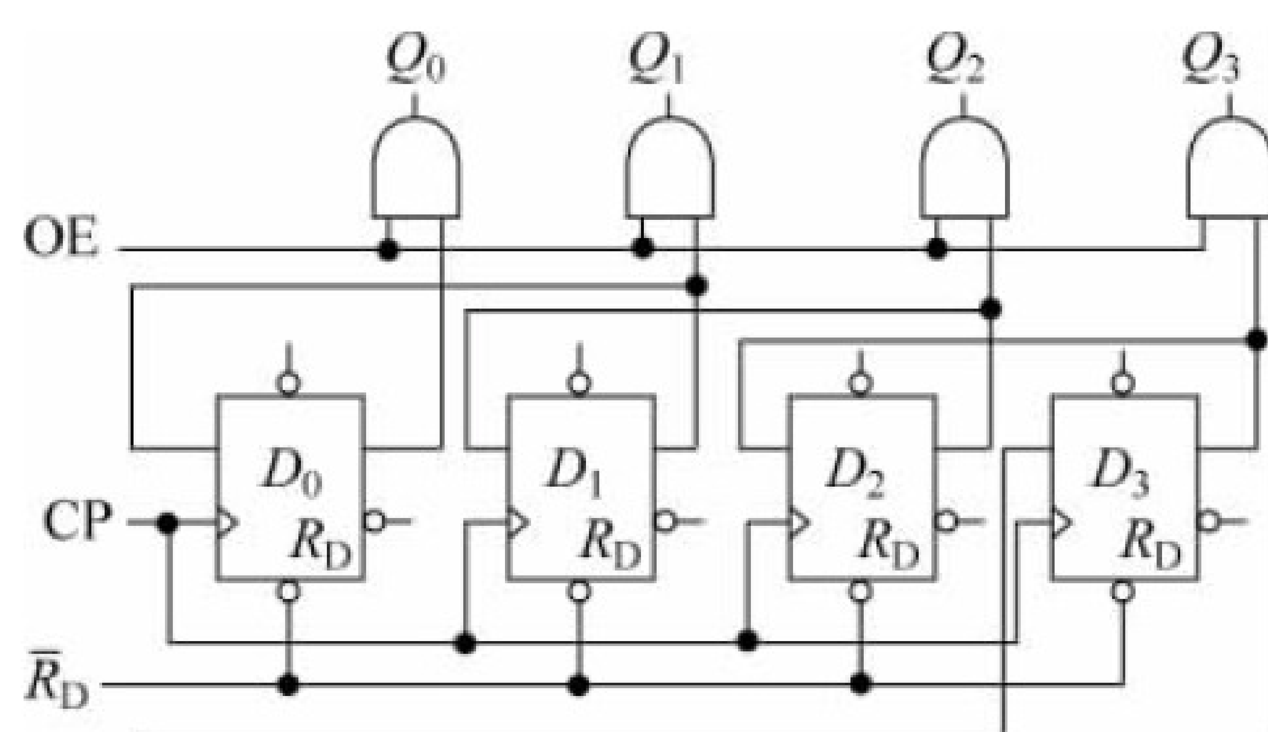
题图 10.5

9. 用 Verilog HDL 的门级结构描述方法描述如题图 10.6 所示的电路，图中的 JK 触发器如例 10.3.2 所示，已经定义好。



题图 10.6

10. 用 Verilog HDL 的门级结构描述方法描述如题图 10.7 所示的电路，图中的 D 触发器如例 10.3.1 所示，已经定义好。



题图 10.7

11. 分析下面代码的逻辑功能。

```
module jksff(j,k,cp,rd,sd,q);
    input j,k,cp,rd,sd;
    output q;
    reg q;
always @(posedge cp or negedge rd or negedge sd)
      begin
        if(!rd) q<=1'b0;
          else if (!sd) q<=1'b1;
          else q<=(j&(~q))|((~k)&q);
      end
endmodule
```

12. 分析下面代码的逻辑功能。

```
module count100(out,en,clk,co);
    output [3:0] out;
    input clk,en;
    output co;
    reg [3:0] out;
    reg co;
always @(posedge clk)
      begin
        if (en) out<=out+1;
      if (out==0) out<=4'd005;
      if (out==15) co<=1;
              else co<=0;
      end
endmodule
```

13. 请分析下面代码的逻辑功能。

```
module count100(out,en,clk,co);
    output [3:0] out;
    input clk,en;
    output co;
    reg [3:0] out;
    reg co;
always @(posedge clk)
```

```
        begin
         if (en) out = out + 1;
        if (out == 0) out = 4'd005;
        if (out == 15) co = 1;
               else co = 0;
        end
endmodule
```

14. 分析下面代码的逻辑功能。

```
module count100(out,data,load,reset,en,clk,co);
   output [7:0] out;
   input [7:0] data;
   input load,reset,clk,en;
   output co;
   reg [7:0] out;
   reg co;
always @(posedge clk)
        begin
         if(!reset)out <= 8'd000;
           else if (!load) out <= data;
             else if (en) out <= out + 1;
        if (out >= 100) out <= 8'd000;
        if (out == 99) co <= 1;
               else co <= 0;
        end
endmodule
```

15. 用 Verilog HDL 描述芯片 74LS161。

16. 用 Verilog HDL 描述芯片 74LS163。



结束语

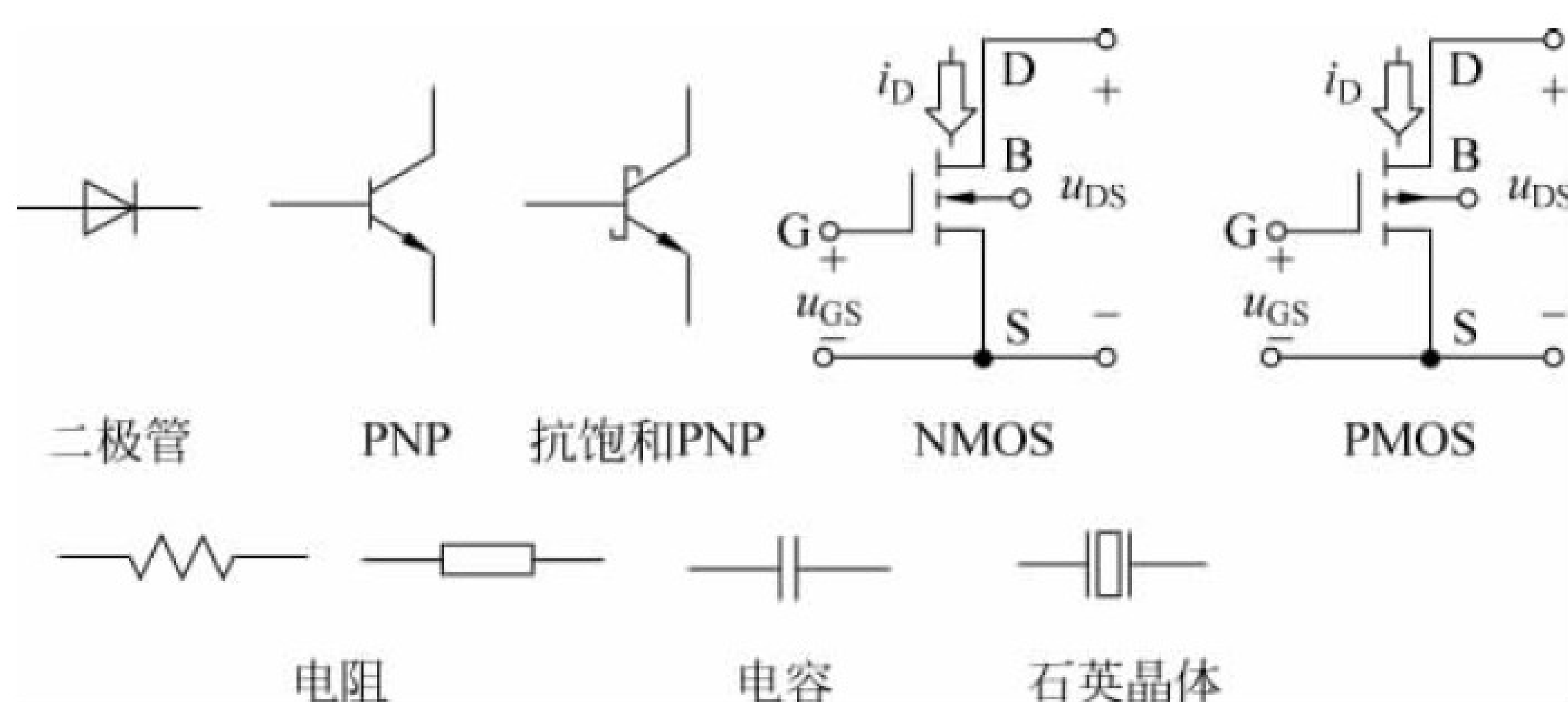


附录A

本书中使用的电路符号



1. 基础器件



2. 门电路

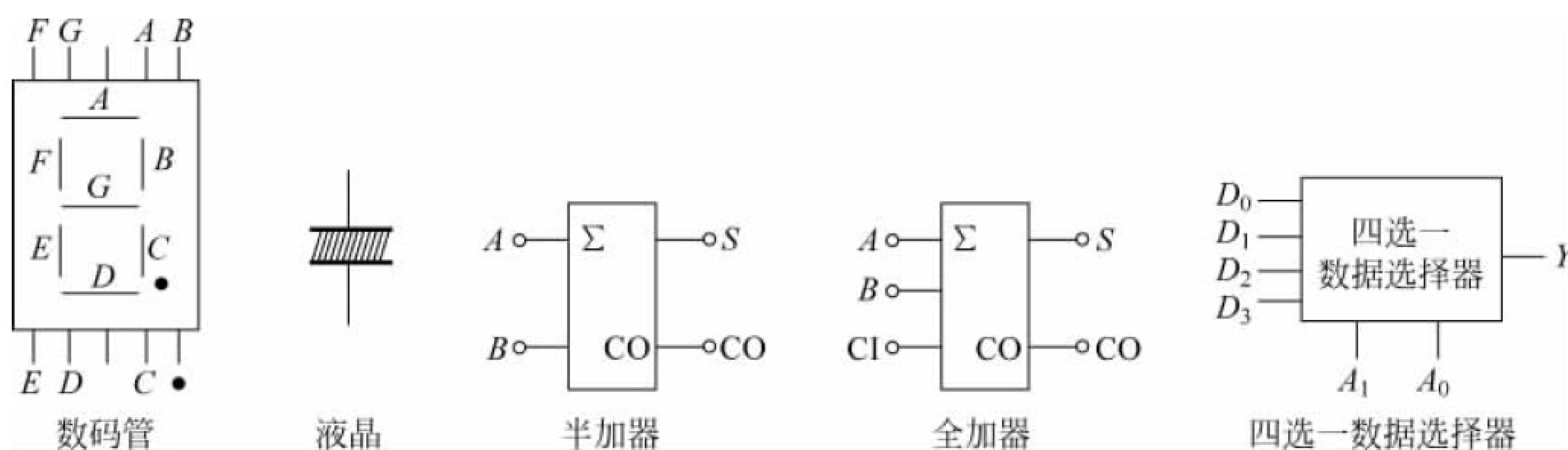
逻辑运算	表　达　式	国 标 符 号	其 他 符 号
或运算	$Y=A+B$	≥1	+
与运算	$Y=A\cdot B$	&	
非运算	$Y=\overline{A}$	1	
与非运算	$Y=\overline{AB}$	&	
或非运算	$Y=\overline{A+B}$	≥1	+
异或运算	$Y=A\oplus B$	=1	⊕

续表

逻辑运算	表 达 式	国 标 符 号	其 他 符 号
同或运算	$Y=A\odot B$		
与或非运算	$Y=\overline{AB+CD}$		

OC门　TS门　CMOS传输门

3．组合器件



数码管　液晶　半加器　全加器　四选一数据选择器

4．触发器

逻 辑 符 号	触发器描述	触 发 方 式	触发器描述
	上升沿 RS 触发器、同步 RS 触发器；不加说明为上升沿 RS 触发器		上升沿 JK 触发器、同步 JK 触发器；不加说明上升沿 JK 触发器
	主从 RS 触发器、下降沿 RS 触发器		主从 JK 触发器、下降沿 JK 触发器
	上升沿 D 触发器、同步 D 触发器；不加说明上升沿 D 触发器		主从 D 触发器、下降沿 D 触发器
(a) (b)	施密特触发器，图(a)电压传输特性如下：		



附录B

本书中介绍的芯片



芯片引脚(或逻辑符号)	简 单 描 述	芯 片 引 脚	简 单 描 述
74LS147	集成二—十进制优先编码器($\overline{I}_9$ 优先级最高);低电平输入有效,反码输出	74LS148	3 位二进制优先编码器,功能与 147 类似。$\overline{S}_T$ 为选通输入端;Y_S 为选通输出端;$\overline{Y}_{EX}$ 为扩展输出端
74LS138	3-8 译码器,低电平输出有效;A_2、A_1、A_0 为译码器的地址端;$\overline{Y}_0 \sim \overline{Y}_7$ 为译码器的输出端。S_T、$\overline{S}_1$、$\overline{S}_2$ 为控制端。当 $S_T=1$、$\overline{S}_1=\overline{S}_2=0$ 时,译码器工作,有 $\overline{Y}_i=\overline{m}_i$。可用于实现组合电路		
74LS42	二—十进制译码器,低电平输出有效;A_3、A_2、A_1、A_0 为译码器的地址输入端;$\overline{Y}_0 \sim \overline{Y}_9$ 为译码器的输出端。 $\overline{Y}_i=\overline{m}_i(i<10)$		

续表

<table>
<tr><th>芯片引脚(或逻辑符号)</th><th>简 单 描 述</th><th>芯 片 引 脚</th><th>简 单 描 述</th></tr>
<tr><td>74LS153
A_1 A_0
S_1 D_{10} D_{11} D_{12} D_{13} Y_1
S_2 D_{20} D_{21} D_{22} D_{23} Y_2</td><td colspan="3">A_0A_1为两个四选一数据选择器的公共地址输入端。$D_{10}\sim D_{13}$、Y_1为上面的四选一数据选择器的输入输出；
$D_{20}\sim D_{23}$、Y_2为下面的四选一数据选择器的输入输出；
外部输入$\overline{S}_1$、$\overline{S}_2$ 为两个四选一数据选择器的使能端</td></tr>
<tr><td>74LS48
7 A_0, 1 A_1, 2 A_2, 6 A_3, 3 $\overline{\mathrm{LT}}$, 4 $\overline{\mathrm{BI/RBO}}$, 5 $\overline{\mathrm{RBI}}$
A 13, B 12, C 11, D 10, E 9, F 15, G 14</td><td>驱动共阴数码管的七段显示译码器。$\overline{\mathrm{BI}}=\overline{\mathrm{RBI}}=\overline{\mathrm{LT}}=1$ 时正常译码</td><td>74LS283
5 A_1, 3 A_2, 14 A_3, 12 A_4, 6 B_1, 2 B_2, 15 B_3, 11 B_4, 7 CI
S_1 4, S_2 1, S_3 13, S_4 10, CO 9, Vcc 16, GND 8</td><td>超前进位的 4 位全加器。如输入 $A_4A_3A_2A_1$(1000)、$B_4B_3B_2B_1$(0110)、CI(0)，则输出为 S_4 S_3 S_2 S_1(1110)、CO(0)</td></tr>
<tr><td>74LS151
4 D_0, 3 D_1, 2 D_2, 1 D_3, 15 D_4, 14 D_5, 13 D_6, 12 D_7
11 A_0, 10 A_1, 9 A_2, 7 $\overline{S}$
Y 5, $\overline{Y}$ 6, VCC 16, GND 8</td><td colspan="3">八选一数据选择器。$D_0\sim D_7$为 8 路输入信号，A_2、A_1、A_0为选择控制信号，Y、$\overline{Y}$ 为互补的输出端，$\overline{S}$ 为选通控制端。当选通控制端 $\overline{S}=0$ 时，选择器工作，输出 Y 的逻辑表达式为
$$Y = D_0\overline{A}_2\overline{A}_1\overline{A}_0 + \cdots + D_7A_2A_1A_0 = \sum_{i=0}^{7} D_i m_i$$
可用八选一数据选择器实现四变量及以下的逻辑函数</td></tr>
<tr><td>74LS85
A_0 B_0 A_1 B_1 A_2 B_2 A_3 B_3 $I_{A>B}$ $I_{A=B}$ $I_{A<B}$
$Y_{A>B}$ $Y_{A=B}$ $Y_{A<B}$</td><td colspan="3">$A_3A_3A_1A_0$、$B_3B_2B_1B_0$为 2 个 4 位的二进制数；
$Y_{A>B}$、$Y_{A=B}$、$Y_{A<B}$为输出的 3 个比较结果。
根据 4 位二进制数相互比较的逻辑关系，当 $A>B$ 时，$Y_{A>B}=1$；$A<B$ 时，$Y_{A<B}=1$；当 $A=B$ 时，输出结果由来自低位的比较结果输入 $I_{A>B}$、$I_{A=B}$、$I_{A<B}$确定</td></tr>
</table>

续表

芯片引脚(或逻辑符号)	简单描述	芯片引脚	简单描述
74LS195	$\overline{CR}$为清零端；SH/$\overline{LD}$为移位、置数端，控制 $D_0 \sim D_3$、$Q_0 \sim Q_3$ 并行输入、输出数据。 $Q_0 = J\overline{Q}_0^n + \overline{K}Q_0^n$	74LS194	集成双向移位寄存器；M_0、M_1 为工作方式控制端、D_{SL} 为左移串行数据输入端，D_{SR} 为右移串行数据输入端
74LS197	4 位二进制异步加法计数器，CP_0 为触发器 FF_0 的时钟端，CP_1 是 FF_1 的时钟端，称为二—八—十六进制异步计数器	74LS161	具有异步清零、同步置数、可以保持状态不变的 4 位二进制同步上升沿加法计数器。163 与 161 功能相似，区别为 163 采用同步清零
74LS290	异步十进制计数器	74LS191	单时钟集成 4 位二进制同步可逆计数器；$\overline{U}/D$ 为加减计数控制端，$\overline{LD}$为异步置数端态输出端，CO/BO 为进位/借位信号输出端，$\overline{RC}$为级联端
74LS193	CR 为异步清零端；$\overline{LD}$为异步置数端；CP_U为加法计数脉冲端；CP_D为减法计数脉冲端；$\overline{CO}$为进位端；$\overline{BO}$为借位端	F007	通用型运放，分析时一般把它当成理想运算放大器
555	555 定时器，应用十分广泛，$\overline{R}_D=1$ 工作。 当 $U_{I1}<U_{R1}=\frac{2}{3}U_{CC}$、$U_{I2}>U_{R2}=\frac{1}{3}U_{CC}$时，输出 U_O状态不变 当 $U_{I1}>U_{R1}$、$U_{I2}<U_{R2}$时，输出 $U_O=1$ 当 $U_{I1}>U_{R1}$、$U_{I2}>U_{R2}$时，输出 $U_O=0$ 当 $U_{I1}<U_{R1}$、$U_{I2}<U_{R2}$时，输出 $U_O=1$		

续表

芯片引脚(或逻辑符号)	简单描述	芯片引脚	简单描述
	A_2、A_1为两个负脉冲触发输入端，B为正脉冲触发输入端，U_O为单稳态触发器输出端，U'_O为反相输出端，R、C为外部电阻、电容接入端，R_{in}为内部电阻连接端。 典型应用电路如下：		
	十位D/A转换器，内部没有运算放大器；U_{DD}为CMOS开关工作电源，U_{REF}为转换器的参考电压，I_{OUT1}、I_{OUT2}分别对应外接运算放大器的反相端及同相端。典型接法的转换关系为 $U_O=-\frac{U_{REF}}{2^{10}}\sum_{i=0}^{9}D_i\times 2^i$		
	$A_9\sim A_0$为地址线；$D_3\sim D_0$为数据线；容量为1K×4；$\overline{CS}$为使能端；R/$\overline{W}$为读写控制端		集成采样—保持电路。当$u_L=1$时，采样，$u_O=u_I$；当$u_L=0$时，保持。U_+、U_-为电源输入端。U_B为偏置电压输入端
	• 具有11根地址线$A_{10}\sim A_0$、8根数据线$D_7\sim D_0$； • 输出使能控制端（当$\overline{OE}=0$时，存储单元内容允许输出）； • 片选控制（当$\overline{CS}=0$时，芯片工作）； • 专用设备擦除、专用设备写入。 • 除可实现存储功能外，尚可实现组合电路。 • 如果将地址端作为输入变量，将数据线作为输出变量，适当地选择存储单元内容，则可实现能为用户编程的组合逻辑电路		

续表

芯片引脚(或逻辑符号)	简 单 描 述	芯 片 引 脚	简 单 描 述
ADC0809 11 U_{DD} 26 IN_0 … 27, 28, 1, 2, 3, 4, 5 IN_7 16 $U_{REF(-)}$ 12 $U_{REF(+)}$ 9 EOUT 6 START 10 CP 13 GND 21 D_7 … 20, 19, 18, 8, 15, 14, 17 D_0 7 EOC 25 A 24 B 23 C 22 ALE	8 路输入 8 位逐次逼近型 A/D 转换器。 IN_0～IN_7：8 路模拟量输入端； A、B、C：8 路模拟量输入选择控制端； ALE：地址锁存输入端，高电平有效，可加正脉冲； D_0～D_7：8 路数字量输出端； EOC：转换结束输出端，高电平有效； EOUT：输出允许端，高电平有效； START：转换启动信号输入端，可加正脉冲，上升沿转换器清零，下降沿开始转换； CP：外部时钟输入端，典型频率为 640kHz； $U_{REF(-)}$、$U_{REF(+)}$：转换器参考电源输入端		



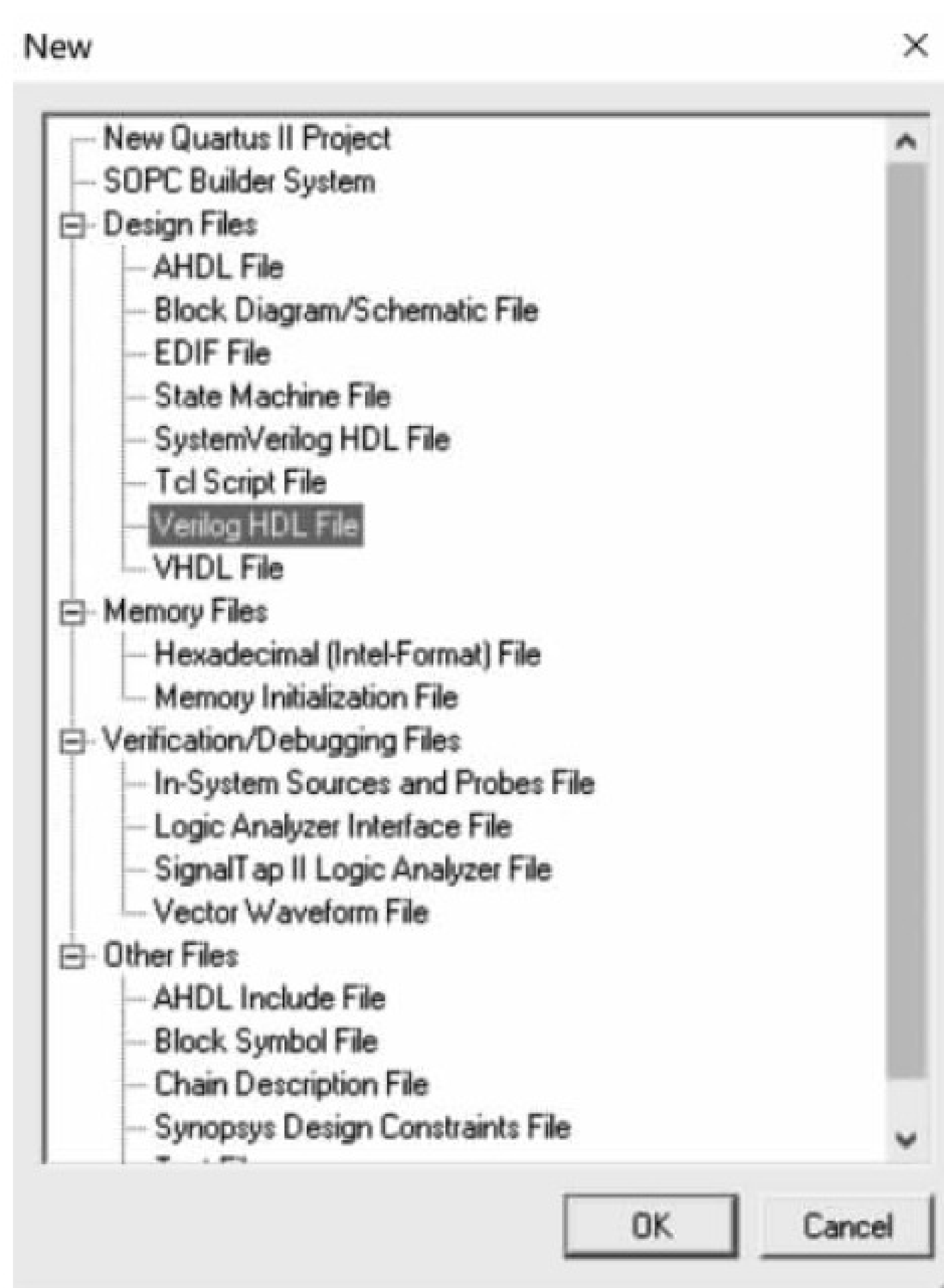
附录C

Quartus Ⅱ中例3.5.1的仿真实现方法

下面利用 Quartus Ⅱ 9.0 SP2 介绍例 3.5.1 的仿真实现方法。

1. 建立仿真项目的工程文件

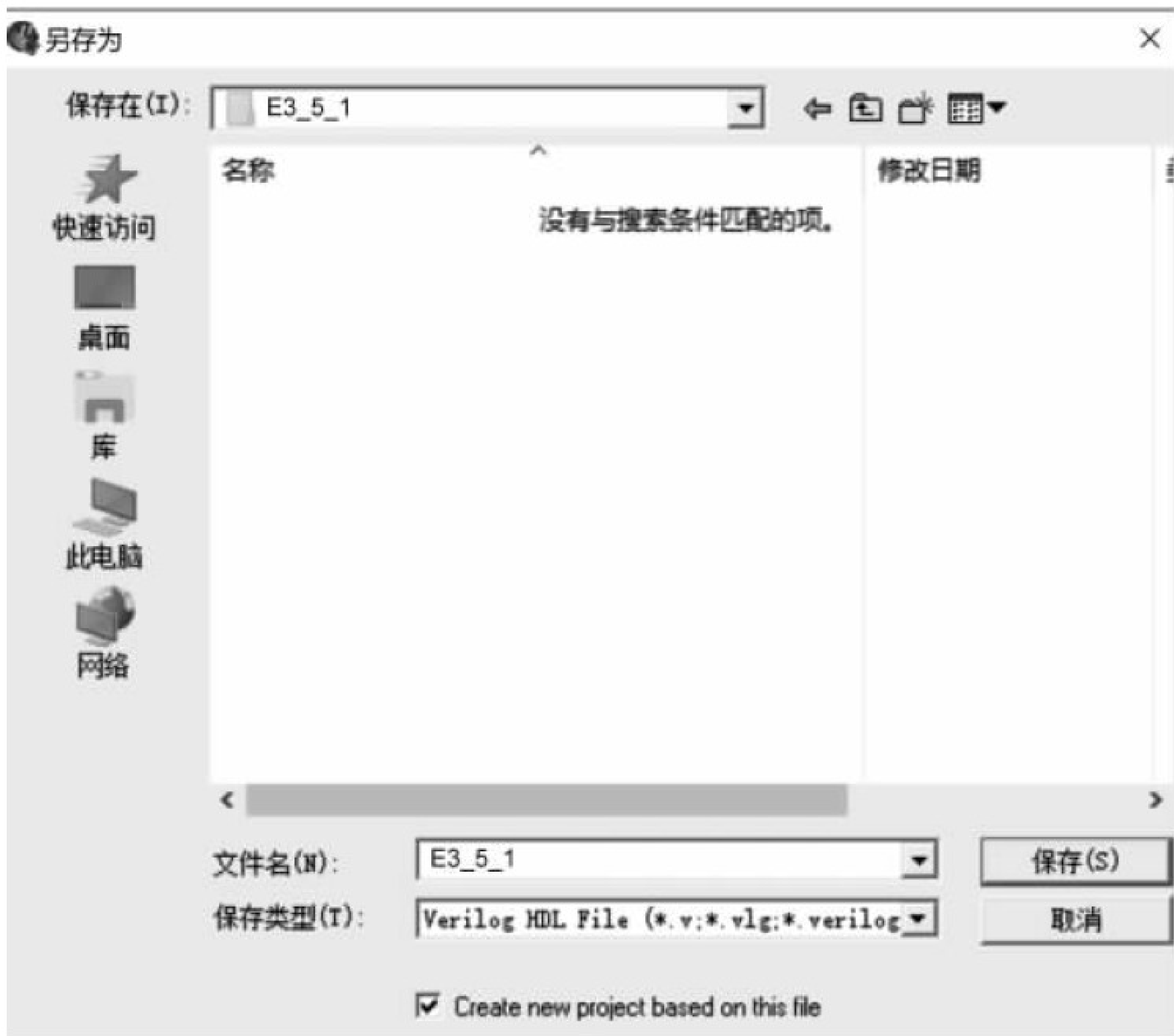
启动 Quartus Ⅱ，选择 File→new 子菜单，在随后弹出的任务窗格中选择文件类型为 Design Files→Verilog HDL File，参考界面如图附 C.1 所示，单击 OK 按钮进入 Verilog HDL 代码编辑界面。



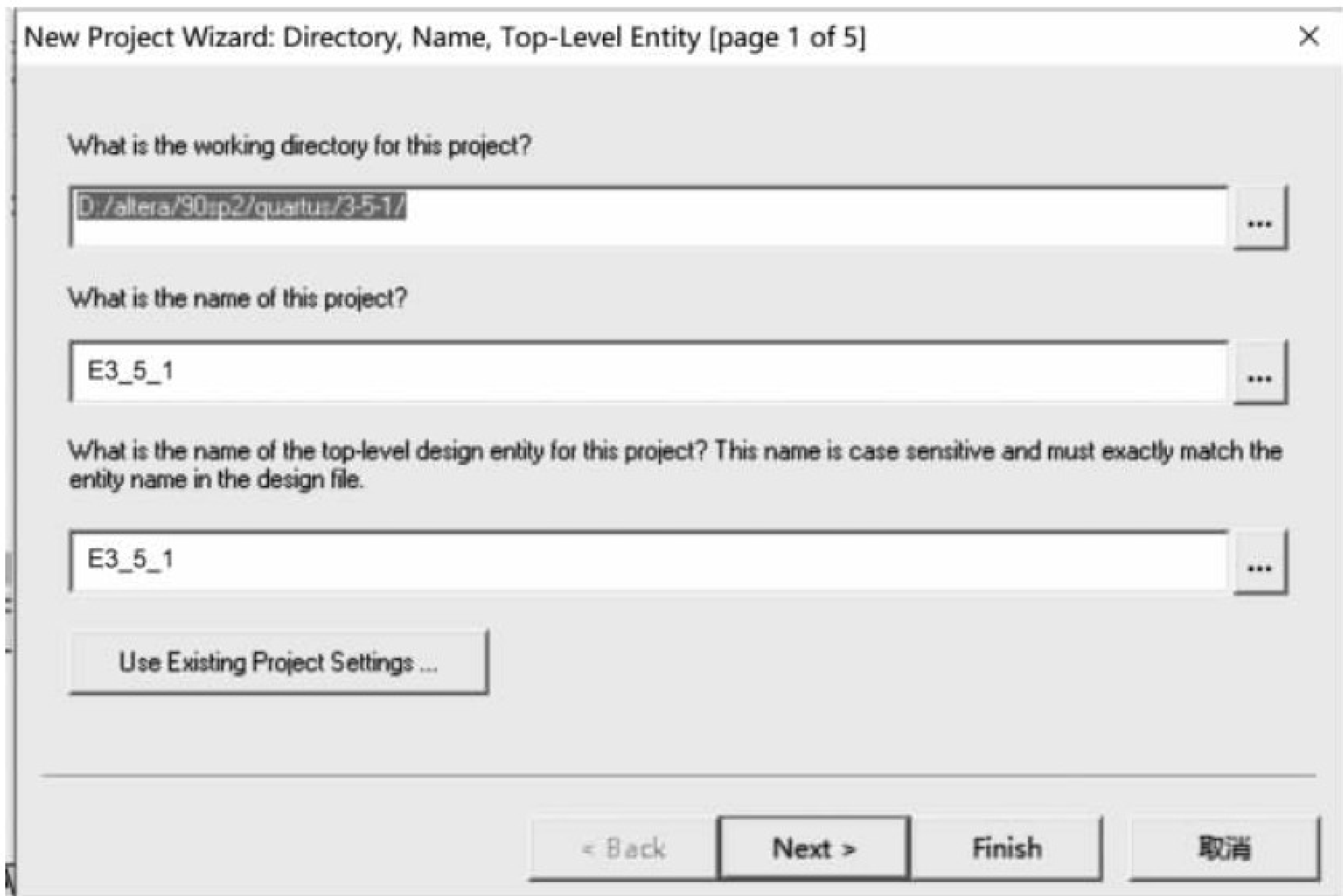
图附 C.1　新建 Verilog HDL File

选择 File→Save As 子菜单，在随后弹出的“另存为”窗口中为新创建的未命名的图形文件取适当的名字(如 E3_5_1。因 Quartus Ⅱ仿真时要产生文件，因此最好为仿真项目新

建一个子目录)，单击“保存”按钮。确认保存前注意勾选任务窗口最下方的 Create new project based on this file 复选框，如图附 C.2 所示。在随后弹出的“创建工程文件确认对话框”中单击“是”按钮进入工程创建界面，如图附 C.3 所示。在该界面中，可选择工作目录，也可直接单击 Finish 按钮完成工程文件的创建。



图附 C.2 另存 Verilog HDL File

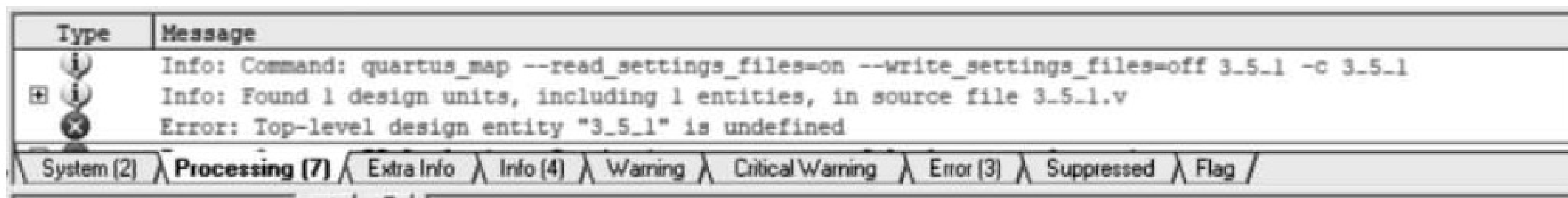


图附 C.3 工程创建界面

2. 输入 Verilog HDL 代码并编译

在编辑区输入代码并保存。保存设计后,选择 Processing→Start Compilation 子菜单编译电路,系统提示出错。

当编译有错误时,可选择工作区最下方的 message 区域,适当移动最右方的滚动条,查看编译错误的提示,如图附 C.4 所示。

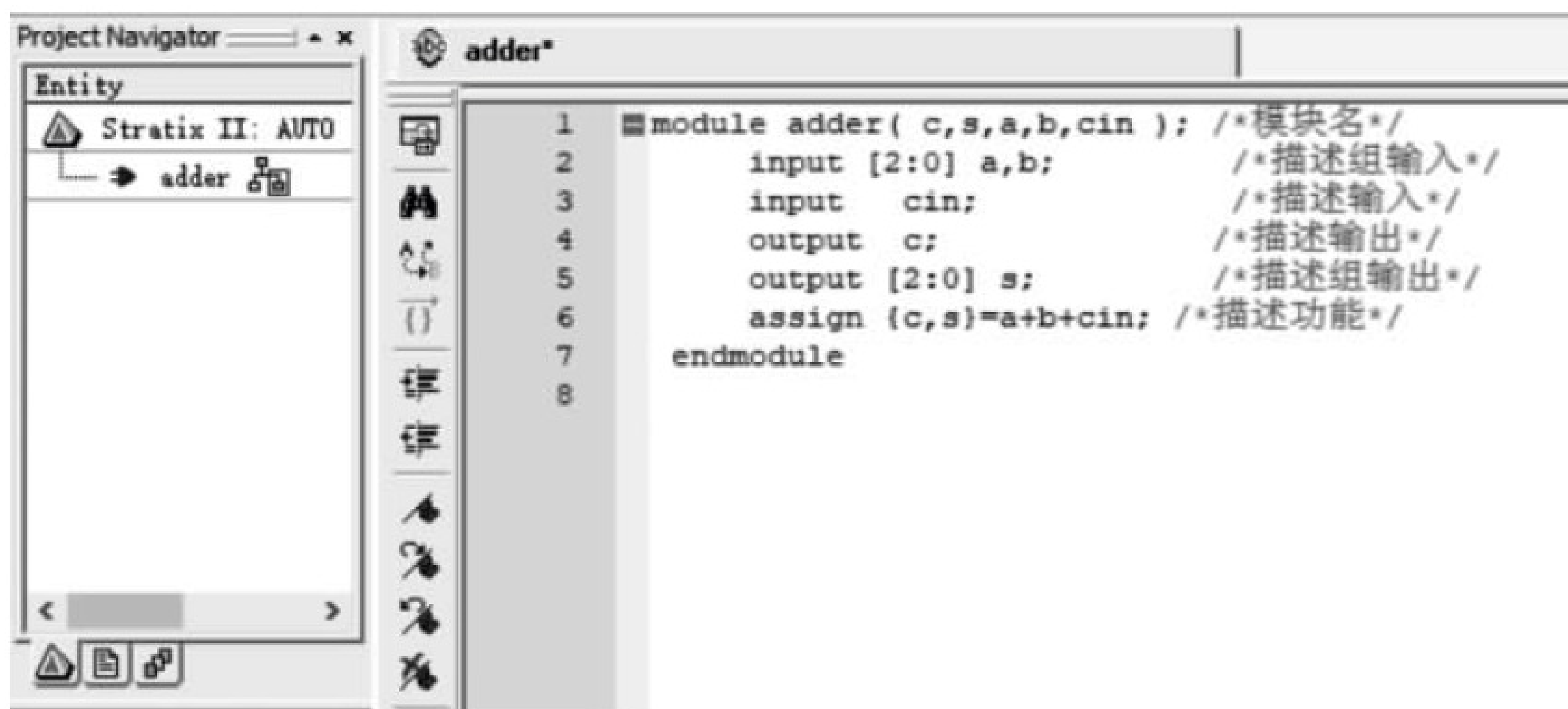


图附 C.4 编译错误的提示界面

Quartus Ⅱ中要求模块名与文件名相同,上面的错误提示为电路模块 E3_5_1 没有定义。代码中定义的模块名为 adder。双击 Project Navigator 窗口中的文件 E3_5_1,重新进入代码编辑界面,将模块名 adder 修改为"E3_5_1",重新编译,依旧提示错误。

读者不要忘记,Verilog HDL 遵循 C 语言语法,标识符以_或字母开头,由_、数字、字母等组成,模块名"3-5-1"不符合 C 语言标识符命名规则。

可重新进入 Quartus Ⅱ,按照 1. 中的步骤另存文件为"adder",并创建工程。编辑区输入代码,参考界面如图附 C.5 所示,重新编译文件。



图附 C.5 重建工程后的参考界面

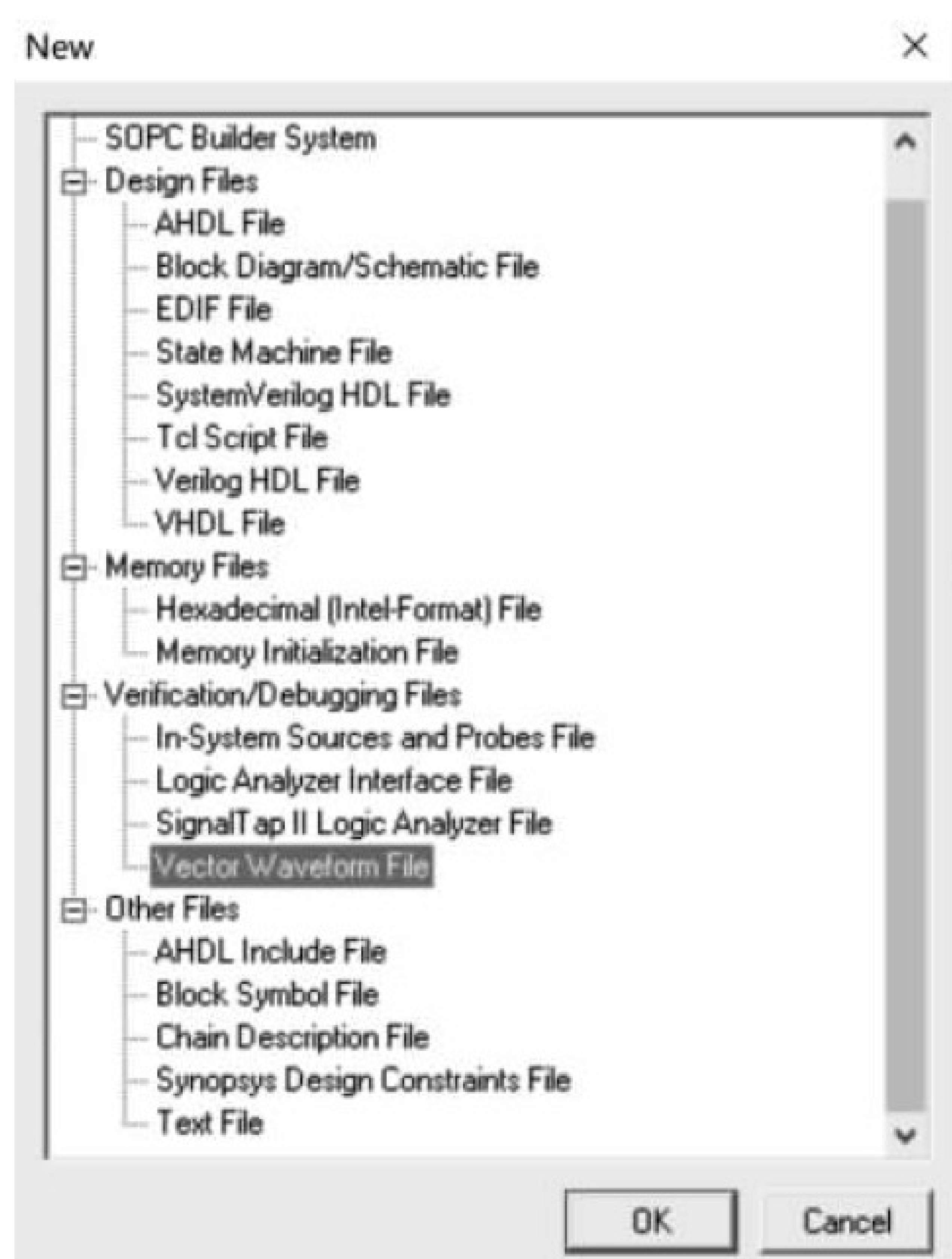
3. 建立要仿真的逻辑图形文件的波形文件

波形文件是 Quartus Ⅱ仿真的必需文件,其主要作用是定义各输入信号及要观察的输出信号。具体实现方法如下:

1) 新建波形文件并添加到当前工程中

选择 File→new 子菜单,在弹出的任务窗口中选择文件类型为 Verification/Debugging Files→Vector Waveform File,如图附 C.6 所示,单击 OK 按钮进入波形文件编辑状态。选

择 File→Save As 子菜单，将新创建的未命名的波形文件命名（默认与代码文件同名，也必须与代码文件同名），单击 OK 按钮保存。确认保存前注意勾选任务窗格最下方的 Add file to current project 复选框，参考界面如图附 C.7 所示。



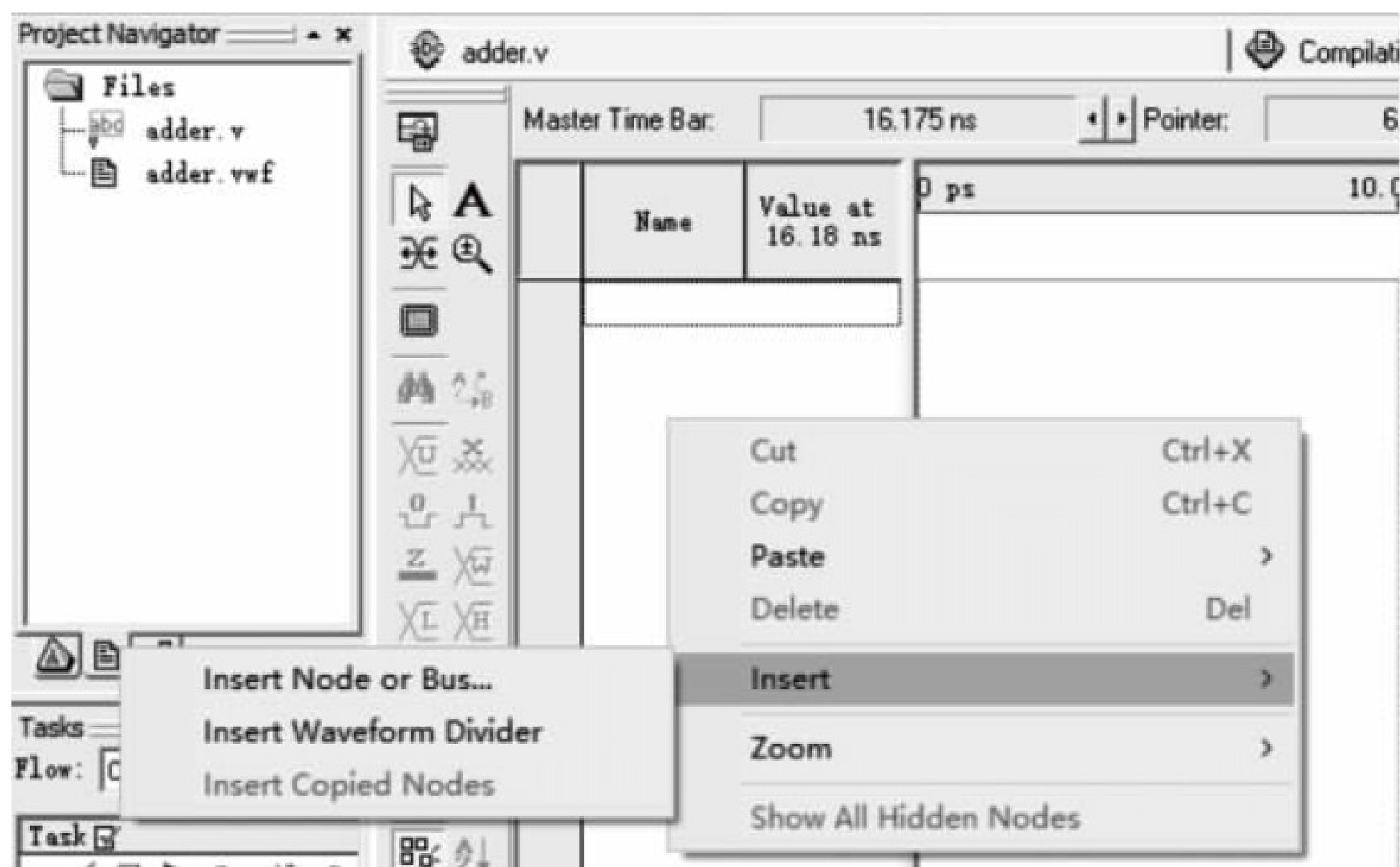
图附 C.6　新建波形文件的图



图附 C.7　另存波形文件的图

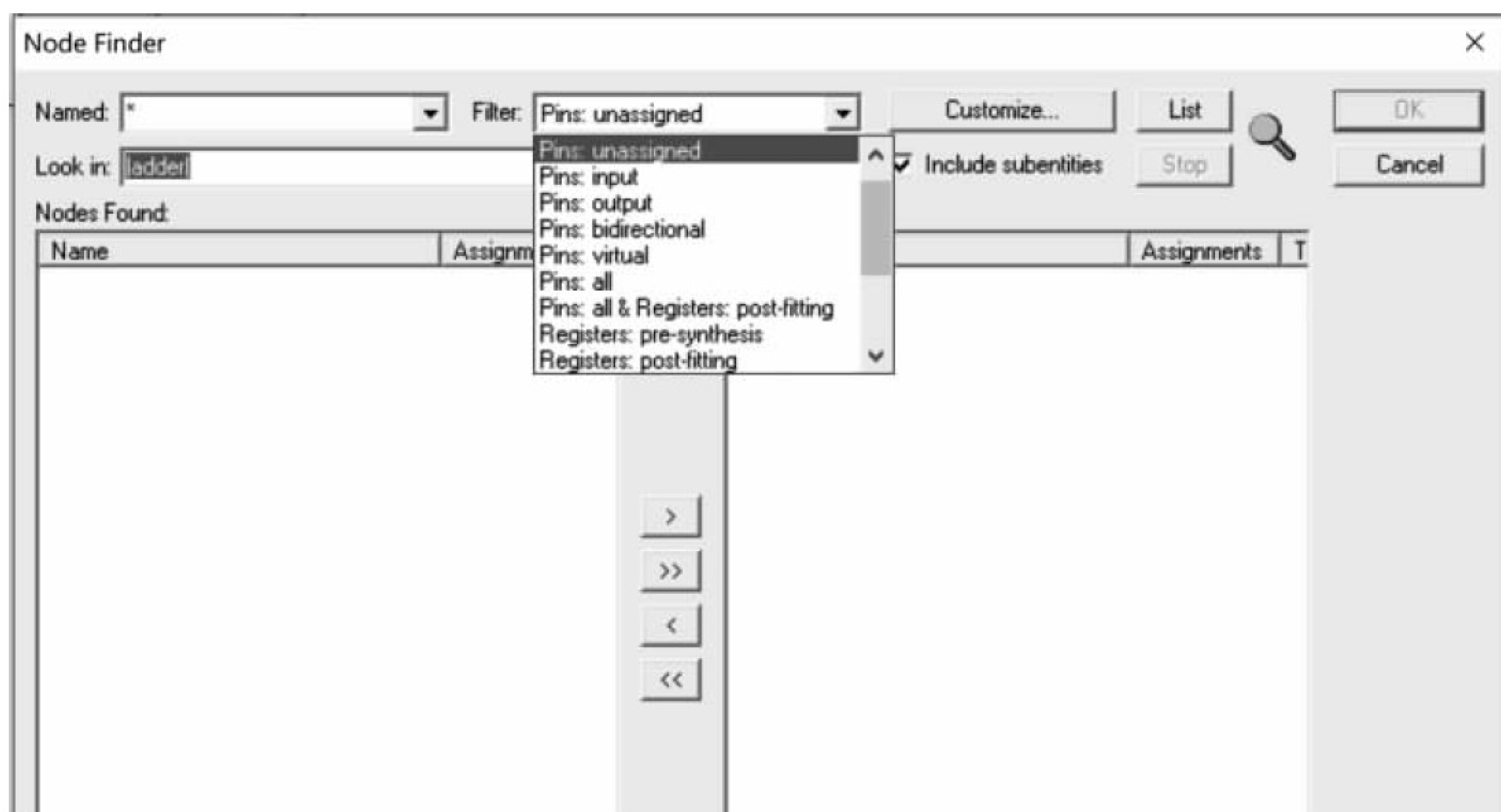
2）导入代码文件中定义的输入输出到波形文件中

在编辑区 name 分栏任意位置右击鼠标，在弹出的菜单中选择 Insert→Insert Node or Bus，如图附 C.8 所示。在随后弹出的对话框中，选择右方的 Node Finder...按钮（右方的第 3 个按钮），将出现 Node Finder 任务窗口。

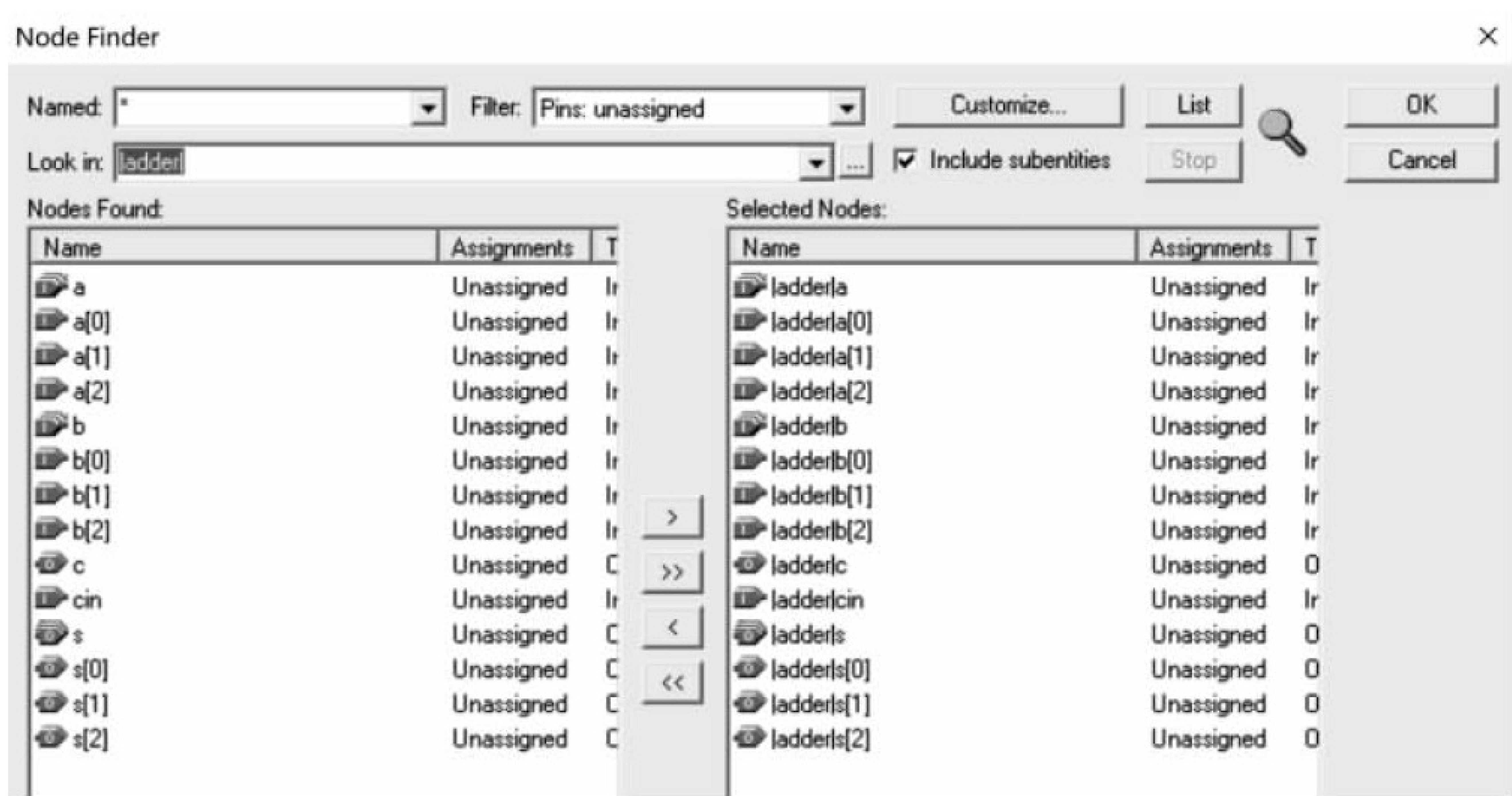


图附 C.8　Node Finder 的图 1

在 Node Finder 任务窗口中，设置上方中间 Filter 下拉框为 Pins：unassigned，参考界面如图附 C.9 所示。单击窗口右上的 List 按钮，在左下文本框中选择想要编辑或观察的信号，单击“>>”按钮，将选择的输入、输出添加到右下文本框，如图附 C.10 所示。



图附 C.9　Node Finder 的图 2

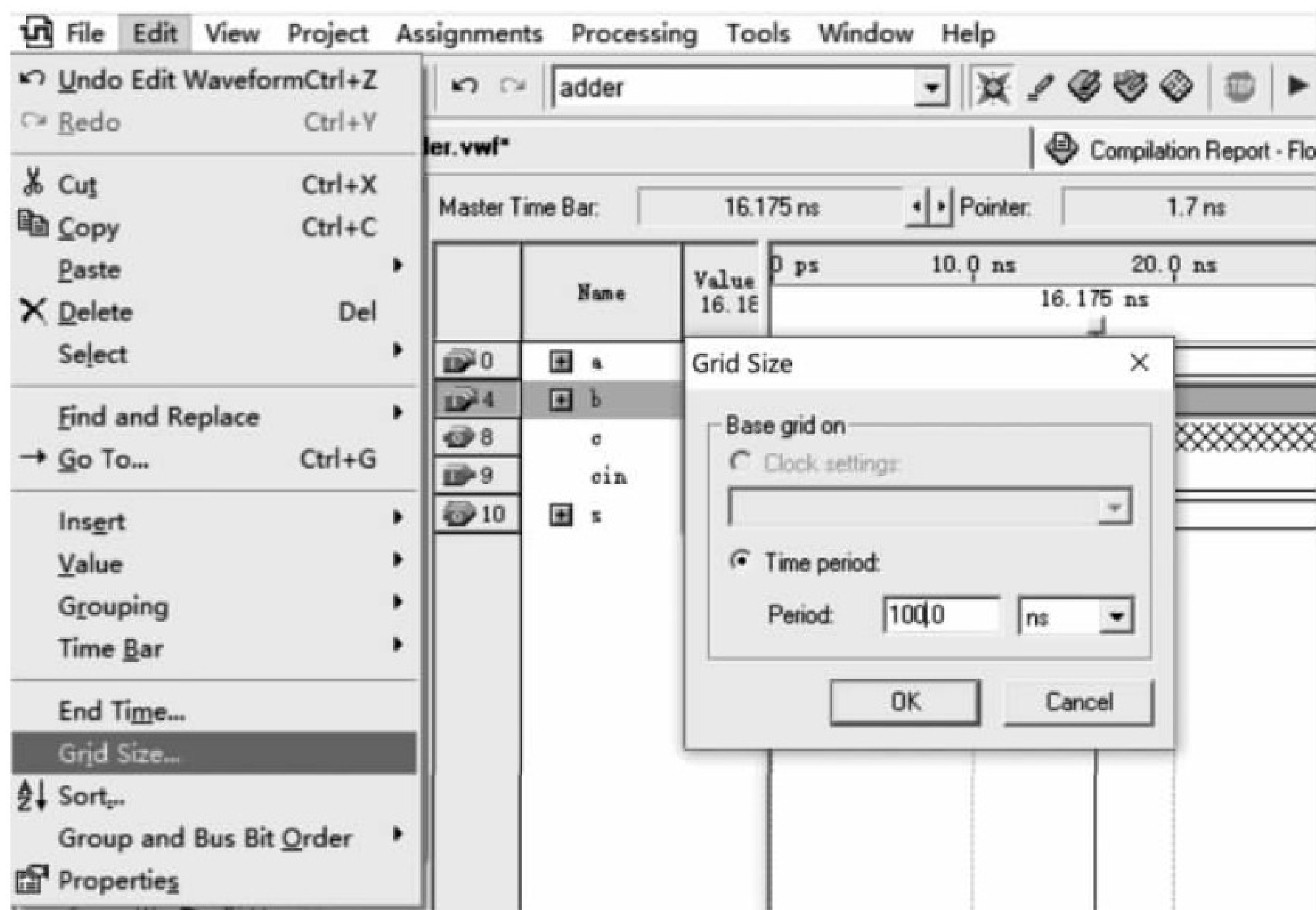


图附 C.10 Node Finder 的图 3

单击 OK 按钮确认，单击 OK 按钮回到波形编辑界面，完成图形文件中的输入输出到波形文件的导入。

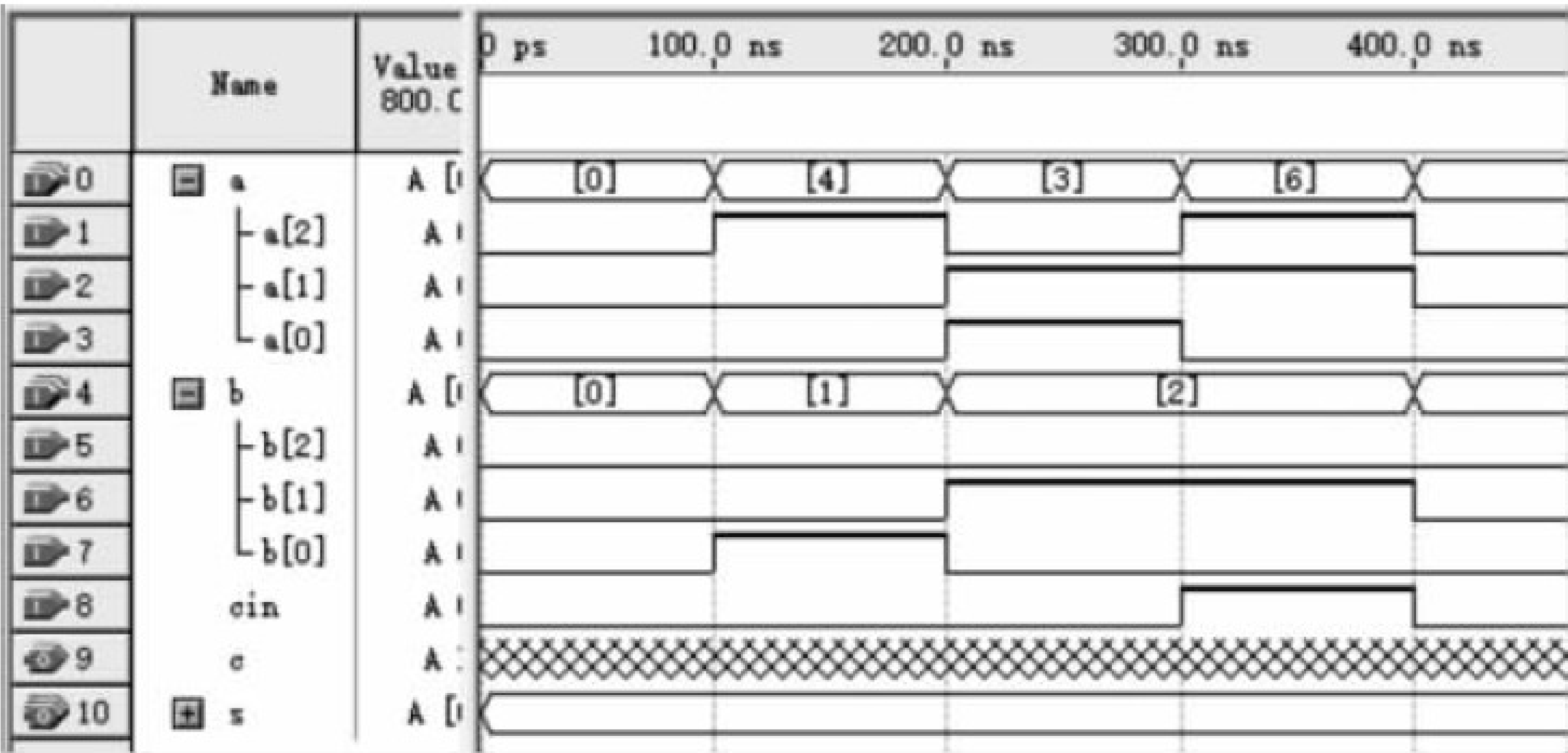
3）设置 Grid Size 及两个 3 位二进制数 a、b，进位 c 的值

选择菜单 Edit→Grid Size，将 Grid Size 的值设置为 100ns，如图附 C.11 所示，重新编译工程。



图附 C.11 设置 Grid Size 的值

拖动鼠标选择要设置的输入的具体的区域，单击左边工具栏中的“0”或者“1”，设置成相应的电平，如图附 C.12 所示。



图附 C.12　参考设置

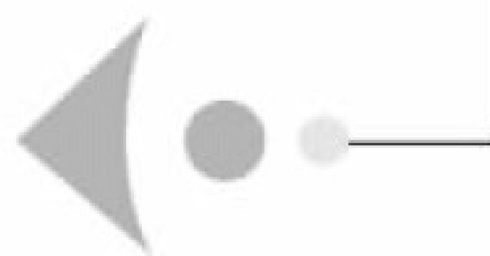
4. 仿真

保存设计后，选择 Processing→Start Simulation 子菜单（第一次仿真前需先单击菜单 Assignment 中的 Settings，在 Settings 窗的 Category 下选 Simulator Settings，在 Simulation Mode 中选择仿真模式：时序仿真 Timing 或功能仿真 Functional；在 Simulation input 中选择 xxx.vwf 仿真波形文件，然后单击 OK 按钮），如果没有错误，系统将弹出仿真成功消息框，求出图 3.5.1 的仿真波形。



附录D

本书仿真包的使用说明



Altera 公司的 Quartus Ⅱ是一个高度集成的可编程逻辑器件开发系统，是目前较为流行的 EDA 软件之一。在本书中，主要利用 Quartus Ⅱ分析组合逻辑电路、时序逻辑电路，验证 Verilog HDL 代码描述的数字系统的逻辑功能，以帮助读者更好地掌握数字电路理论。

为方便读者学习，可扫码该章中本章要点的二维码下载该章相应的仿真源程序包。程序包为自解压缩包，安装到硬盘后启动 Quartus Ⅱ，打开工程文件，选择 Processing→Start Simulation 子菜单即可仿真。

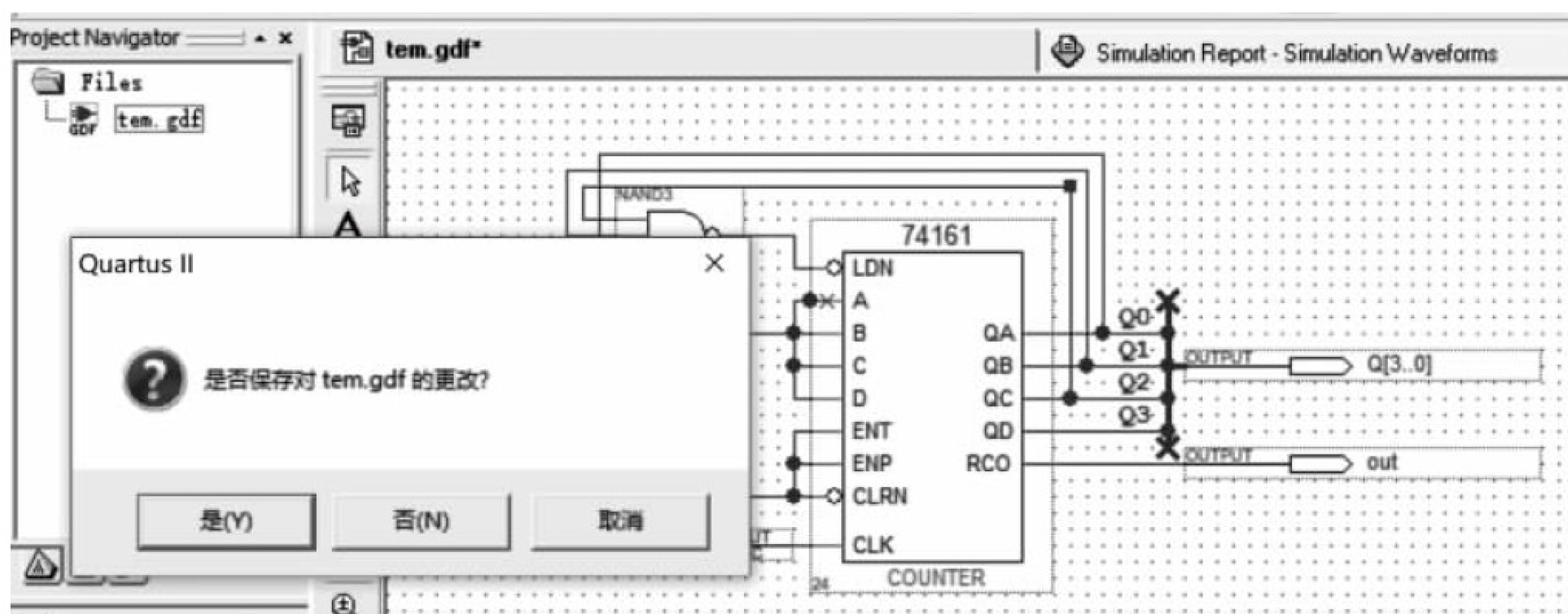
也可直接与本书主编联系（QQ：422260250），索取全书的仿真源程序包。该包中各压缩包命名方法如下：对例题的仿真源程序包，采用章、节、题号命名。如第 3 章第 5 节第一题为 3-5-1。对教材中电路图的仿真与图的编号一致，为与例题仿真源程序包的名字区别，以 T 为前缀，如图 4-3-5，其对应仿真源程序包名为 T4-3-5。

可通过修改电路模型文件实现其他电路模型的仿真分析。本书仿真包中的电路模型文件包括 Verilog HDL 代码文件及图形文件两大类。

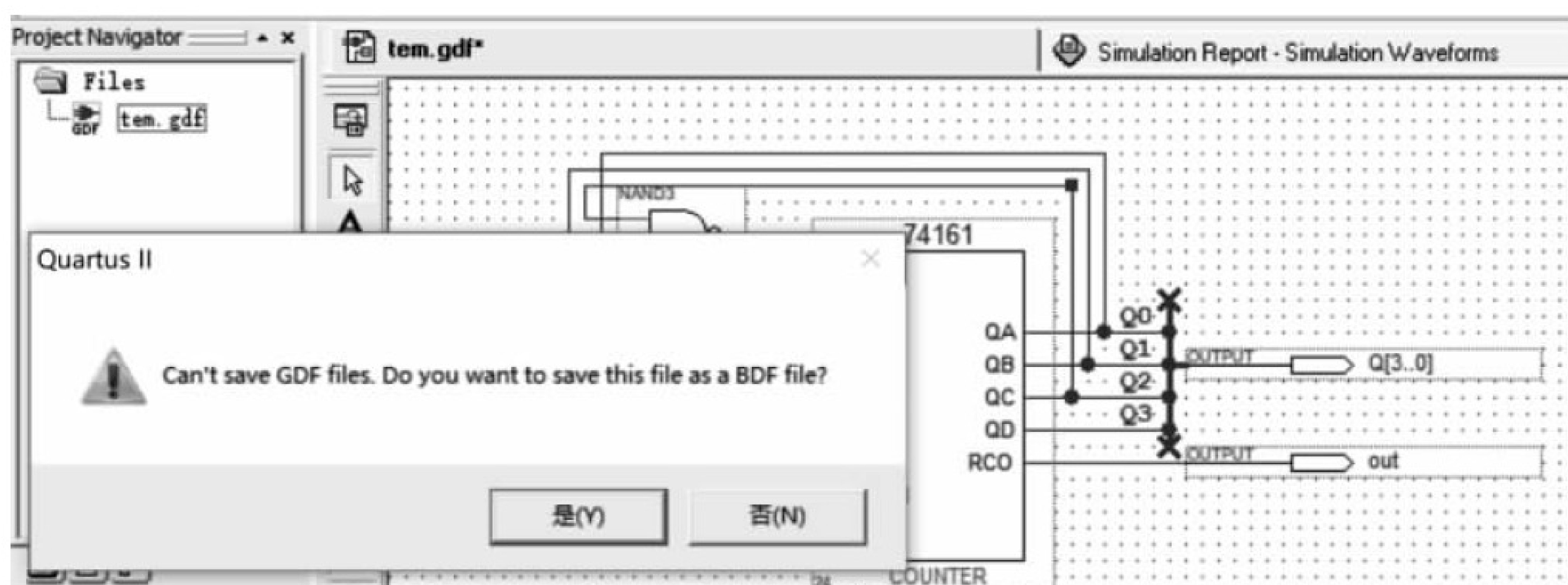
对 Verilog HDL 代码文件，可启动 Quartus Ⅱ，打开工程文件。双击 Project Navigator 窗口中的 Verilog HDL 代码文件进入代码编辑界面，保持模块名不变，修改代码内容，重新编译电路。如果模块中的输入、输出没有修改，可直接选择 Processing→Start Simulation 子菜单求出仿真结果。如果模块中的输入、输出也进行了修改，可按照附录 C 的方法导入代码文件中定义的输入输出，设置输入后仿真即可。

对通过图形文件建立的电路模型，本书各仿真包为基于 MAX＋plus Ⅱ的仿真包，之后，通过 Quartus Ⅱ的工程转换功能转换为 Quartus Ⅱ仿真工程包的，图形文件类型为 gdf。

可修改电路的连接关系，重新编译电路，系统将提示保存修改，如图附 D.1 所示。单击“是”按钮保存修改，系统将提示早期格式不支持，提醒另存为 bdf 格式，参考界面如图附 D.2 所示。



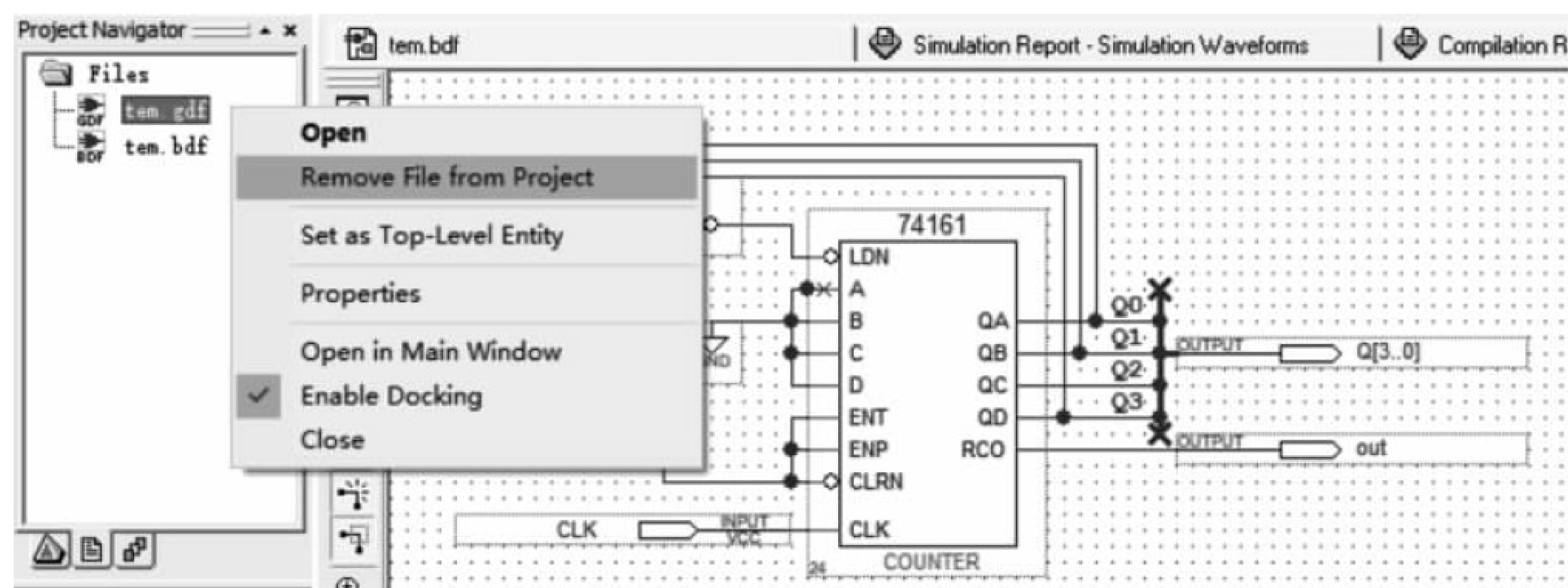
图附 D.1　保存修改提醒对话框



图附 D.2　早期格式不支持，提醒另存为 bdf 格式对话框

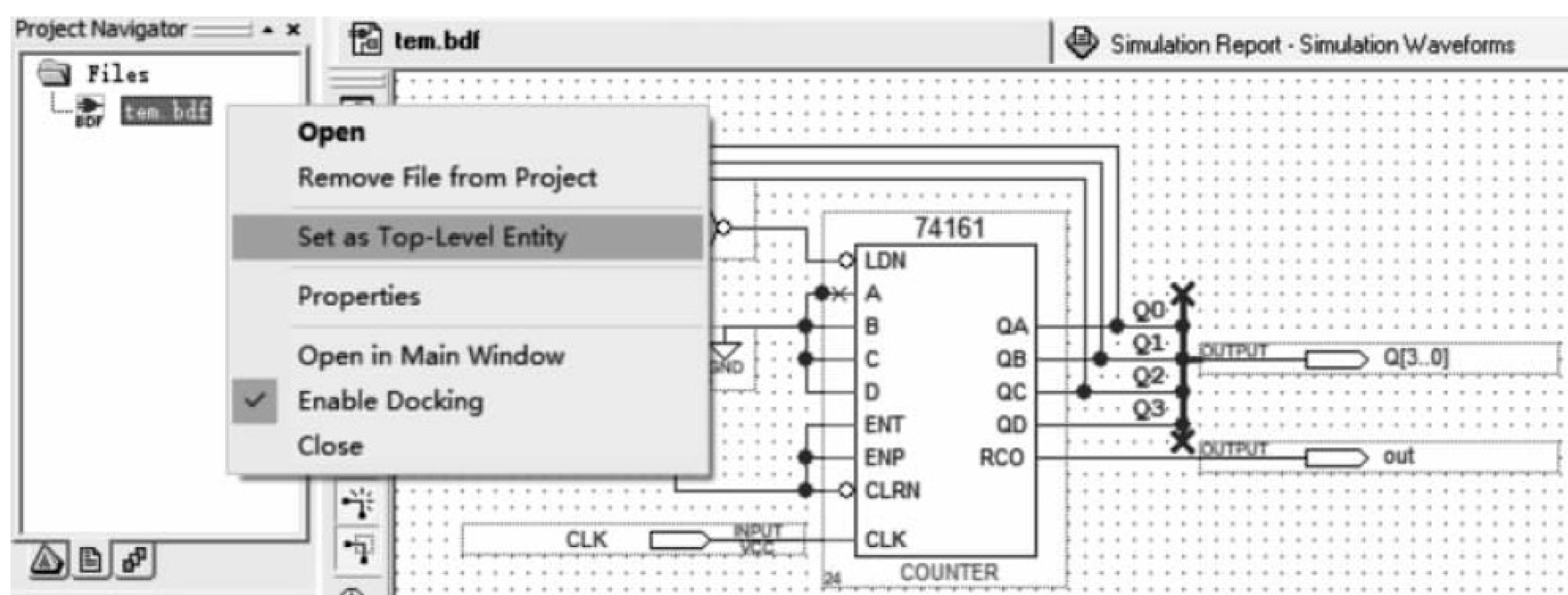
单击“是”按钮保存修改。此时，工程中有 2 个相同文件名的电路模型文件(一个后缀为 gdf，另一个后缀为 bdf)，直接编译会出错。

单击 Project Navigator 任务窗口下方的按钮，切换到 Files 窗口，将鼠标指向 gdf 格式图形文件，单击右键，在弹出的菜单中选择 Remove File from Project，将 gdf 格式模型移除工程，如图附 D.3 所示。



图附 D.3　将 gdf 格式模型移除工程

在弹出的菜单中选择 Set as Top-Level Entity，将 bdf 格式模型设为顶层，如图附 D.4 所示。



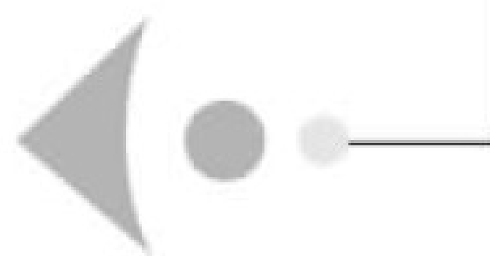
图附 D.4 将 bdf 格式模型设为顶层

编译电路，仿真并求出仿真结果。

如果在修改电路中修改了电路的输入输出，则可按照附录 C 的方法创建波形文件，导入电路中定义的输入输出，设置输入后仿真即可。



附录E



利用本书资源实施翻转课堂教学的建议方法

本书教学视频学习效果在实践中得到了验证，配套资源丰富，非常适合以翻转课堂教学形式开展教学活动，建议选用本书的教师采用传统课堂结合翻转课堂模式开展教学。

具体组织上，可对本书第1、2章采用传统教学模式开展教学。对第4～6章等重点内容采用翻转课堂模式开展教学。

“卡通说解数字电子技术”MOOC中的在线习题丰富，重点内容各知识点的习题量符合采用课内课外颠倒式翻转方法开展教学的要求。对翻转课堂的教学知识点，可采用课内课外颠倒式翻转方法开展教学。课内主要采用在线习题及课外习题、讨论等方式开展教学，也可将部分知识点改由学生主持开展教学。

学生主持的利用中规模器件实现组合逻辑电路的教案如下：

利用中规模器件实现组合逻辑电路

1. 学习目标

掌握层次化和模块化的设计方法，具体为用译码器实现组合逻辑电路、用数据选择器实现组合逻辑电路（表达式对照、卡诺图对照）。

理解组合逻辑电路中的竞争-冒险现象及其原因；会判断是否存在竞争-冒险现象；知道消除竞争-冒险现象的三种方法。

2. 课内教学安排

（1）用译码器实现组合逻辑电路（20分钟）

① 网上视频学习：7分钟。

② 在线习题1：2分钟。

③ 课堂讨论题：6分钟。

唐**同学（input：C）追求刘**同学（input：A）的逻辑问题求解。

逻辑抽象如下：

用 1 表示刘 ** 同学同意，用 0 表示刘 ** 同学不同意；若刘 ** 不同意，则唐 ** 同学霸王硬上弓；刘 ** 同学有一瓶防狼喷雾(用 B 表示)，1 表示防狼喷雾无效，此时也视为追求成功，0 表示防抗喷雾有效；其他情况均视为追求失败；设输出 Y 为"唐 ** 追求成功"，试用 3-8 译码器 74LS138 实现该输出。

④ 在线习题 2：5 分钟。

(2) 数据选择器实现组合逻辑电路(卡诺图对照)(25 分钟)

① 网上视频学习：7 分钟。

② 在线习题 1：2 分钟。

③ 在线习题 2：5 分钟。

④ 讨论题，用八选一数据选择器产生逻辑函数：5 分钟。

$$Y=AC+A'B'C'+A'B'C$$

⑤ 课后习题 4.22：6 分钟。

课间休息

(3) 数据选择器实现组合逻辑电路(表达式对照)(23 分钟)

① 网上视频学习：7 分钟 30 秒。

② 在线习题 1：2 分钟。

③ 在线习题 2：4 分钟。

④ 总结数据选择器：4 分钟。

⑤ 课堂讨论题：5 分钟 30 秒。

用八选一数据选择器设计一个组合逻辑电路。该电路有 3 个输入逻辑变量 A、B、C 和一个工作状态控制变量 M。当 $M=0$ 时，电路实现"意见一致"功能(A、B、C 状态一致时输出 1，否则输出 0)，而 $M=1$ 时，电路实现"多数表决"功能，即输出 A、B、C 中多数的状态一致。

(4) 组合逻辑电路中的竞争-冒险现象(22 分钟)

① 网上视频学习：5 分钟。

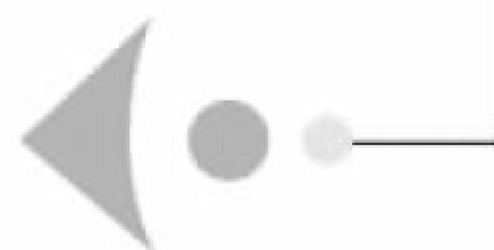
② 在线习题 1：2 分钟。

③ 在线习题 2：3 分钟。

④ 课堂讨论，卡诺图在组合逻辑电路竞争-冒险中的应用：12 分钟。



附录F



部分习题答案

第 1 章

1-2 分析计算题

1. 129_{10},81_{16}；(2) 68_{10},44_{16}；(3) 109_{10},$6D_{16}$；(4) 3.125_{10},3.2_{16}

2. (1) 100101；(2) 110011；(3) 1011100；(4) 1111111

3. 各数从大到小排列为：$(F8)_{16}$,$(302)_8$,$(105)_{10}$,$(1001001)_2$

4. (1) 0.0110110110；(2) 1100001110；(3) 1110111011；(4) 1110000010

5. (1) 381；(2) 344；(3) 6F.1A；(4) 2F.E4

10. (1) 1001111；(2) 01000100；(3) 110001；(4) 1101111

11. (1) 00011011；(2) 11001111；(3) 11000001；(4) 01010011

第 2 章

2-2 分析计算题

1. (1)、(2)、(4)　　3. (1)(×)、(2)(m)、(3)(×)

5. (a) $Y_1=\bar{A}\bar{B}C+\bar{A}B\bar{C}+\bar{A}BC+A\bar{B}\bar{C}+AB\bar{C}+ABC$

   (b) $Y_2=\bar{A}\bar{B}\bar{C}+\bar{A}\bar{B}C+\bar{A}B\bar{C}+A\bar{B}C+AB\bar{C}$

6. (a) $Y_3=\bar{A}\bar{B}\bar{C}D+\bar{A}\bar{B}C\bar{D}+\bar{A}\bar{B}CD+\bar{A}B\bar{C}\bar{D}+\bar{A}BC\bar{D}+\bar{A}BCD+\bar{A}BC\bar{D}+\bar{A}B\bar{C}D+ABC\bar{D}+ABCD$

   (b) $Y_4=\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}\bar{B}\bar{C}D+\bar{A}B\bar{C}\bar{D}+\bar{A}BC\bar{D}+\bar{A}BCD+A\bar{B}\bar{C}D+A\bar{B}C\bar{D}+AB\bar{C}\bar{D}+AB\bar{C}D$

8. (a) $Y=\overline{\overline{ABC}ABC\overline{ABC}}=1$；(b) $Y_2=\overline{ABC}$

9. $Y=\bar{A}\bar{B}C+\bar{A}BC+A\bar{B}\bar{C}+AB\bar{C}$

10. (1) $\bar{A}B+C+\bar{D}$,$A\bar{B}+\bar{C}+D$

(2) $AD+B+\overline{A}\overline{B}+\overline{A}\overline{D}+\overline{B}\overline{C}+\overline{C}\overline{D}$, $\overline{A}\overline{D}+\overline{B}+AB+AD+BC+CD$

(3) $\overline{A}BCD+\overline{B}\overline{C}\overline{D}$, $A\overline{B}\overline{C}\overline{D}+BCD$

(4) $\overline{D}+C[(\overline{A}+\overline{D})\overline{B}+\overline{E}]$, $D+\overline{C}[(A+D)B+E]$

11. (1) $\sum_{m}(3,5,6,7)$；(2) $\sum_{m}(0,1,2,3,9,13,6,7,14,15)$；

(3) $\sum_{m}(2,3,9,10,11,12,13,14,15)$；(4) $\sum_{m}(9,13)$

13. (1)$AB+B\overline{C}$；(2)$A\overline{B}\overline{D}+\overline{B}C\overline{D}$；(3)$A\overline{B}$；(4)$AB$

14. (1) $\overline{B}+C+A\overline{D}$；(2) $AB+C$；(3) $AB+C$；

(4) 包括多个最简结果,其中的 4 个最简结果如下：

$A\overline{B}+B\overline{C}+\overline{A}C+CD$　　　$A\overline{B}+B\overline{C}+\overline{A}C+AD$

$\overline{A}B+\overline{B}C+A\overline{C}+AD$　　　$\overline{A}B+\overline{B}C+A\overline{C}+CD$

(5) $A+BC$

17. (1) B；(2) D；(3) $\overline{B}$；(4) $B\overline{D}+\overline{B}D$；(5) $\overline{A}B\overline{C}+\overline{A}\overline{D}+C\overline{D}+ABC$；

(6) $AB+\overline{B}\overline{C}+A\overline{D}+B\overline{D}$

18. (1) $B+C$；(2) AB；(3) $\overline{A}\overline{B}+C\overline{D}+\overline{B}\overline{D}$；(4) $\overline{D}+\overline{A}\overline{B}+\overline{B}C+AB\overline{C}$；

(5) $C\overline{D}+\overline{A}\overline{B}+\overline{B}\overline{C}$；(6) $A\overline{B}+\overline{B}C+\overline{A}D$

19. (1) $\overline{\overline{AB}\,\overline{BC}}$；(2) $\overline{\overline{A\overline{C}D}\,\overline{\overline{A}\overline{B}}\,\overline{BCD}}$；(3) $\overline{\overline{AB}\,\overline{\overline{B}C}\,\overline{AD}}$；(4) $\overline{\overline{A\overline{C}D}}$

20. (1) $\overline{\overline{AB+BC}}$；(2) $=\overline{\overline{\overline{\overline{A}+C+\overline{D}}+\overline{A+B}+\overline{\overline{B}+\overline{C}+\overline{D}}}}$；

(3) $\overline{\overline{\overline{\overline{A}+\overline{B}}+\overline{B+\overline{C}}+\overline{\overline{A}+\overline{D}}}}$；(4)$\overline{\overline{A}+C+\overline{D}}$

第 3 章

3-2 分析计算题

3. 图(a)：饱和,图(b)：放大。

4. (1) 2V；(2) 3.81V。

5. 5 个(高电平 5 个、低电平 16 个)

6. 5 个(高电平 5 个、低电平 8 个)

7. 0.38kΩ～2.5kΩ 之间

8. 0.68kΩ～5kΩ 之间

11. $Y_1=1$、$Y_2=0$、$Y_3=1$、$Y_4=0$、$Y_5=0$、Y_6高阻、$Y_7=1$、$Y_8=0$

13. (1) $U_I\approx1.4$V；(2) $U_I\approx0.2$V；(3) $U_I\approx1.4$V；(4) $U_I\approx0$V；(5) $U_I\approx1.4$V

14. $Y=A+B$

15. 三态反相器

16. 同或门

17. $Y=AB+\overline{B}C$

第 4 章

4-2 分析设计题

1. $Y=A\overline{B}C$
2. $Y=\overline{ABC}$
3. $Y=\overline{A}B+\overline{C}$
4. $Y=B+\overline{C}$
8. 原始函数如下(更多功能请读者自己分析,下同):

$Z_1(A,B,C)=\sum_m(1,3,5,7)$、 $Z_2(A,B,C)=\sum_m(2,3,6,7)$

$Z_3(A,B,C)=\sum_m(4,5,6,7)$

9. 原始函数如下:

$Z_1(A,B,C)=\sum_m(0,2,4,6)$、 $Z_2(A,B,C)=\sum_m(1,3,5,7)$

10. 原始函数如下:

$Z_1(A,B,C)=\sum_m(0,2,4,7)$、 $Z_2(A,B,C)=\sum_m(1,2,5,7)$

$Z_3(A,B,C)=\sum_m(3,4,5,7)$

11. 原始函数如下:

$Z_1(A,B,C)=\sum_m(0,2,4)$、 $Z_2(A,B,C)=\sum_m(1,2,5)$

$Z_3(A,B,C)=\sum_m(3,4,5)$

12. $Y_3Y_2Y_1Y_0=0110$ $BO=0$;实际含义:$13-7=6$
13. 八选一数据选择器
15. 原始函数如下:

$F(A,B,C)=\sum_m(3,5,6,7)$

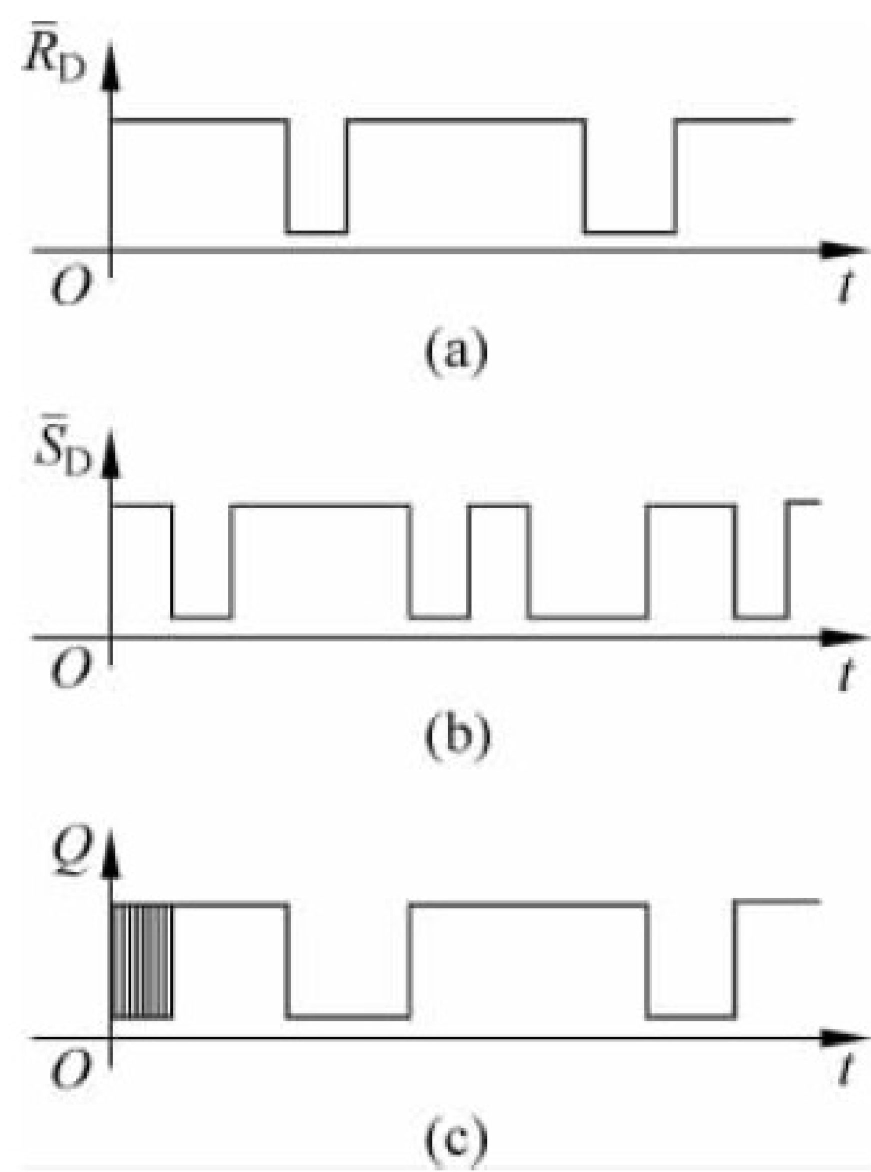
16. 原始函数如下:

$F(A,B,C,D)=\sum_m(0,1,3,4,7,12)$

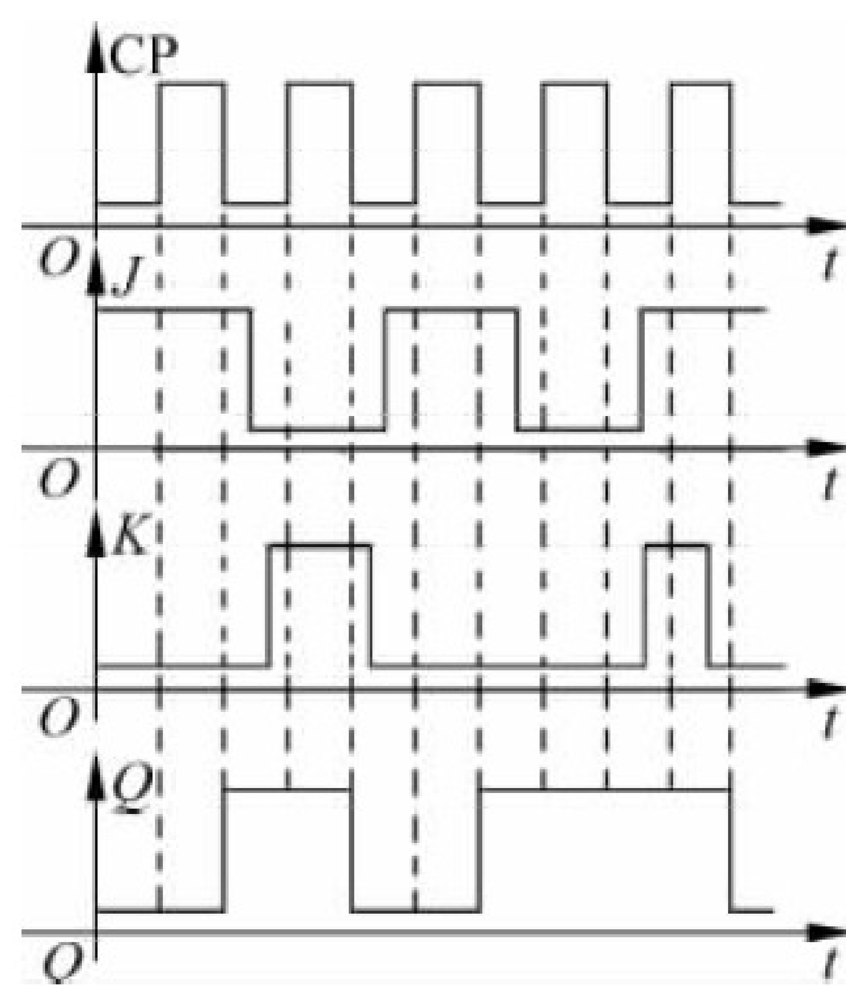
第 5 章

5-2 分析应用题

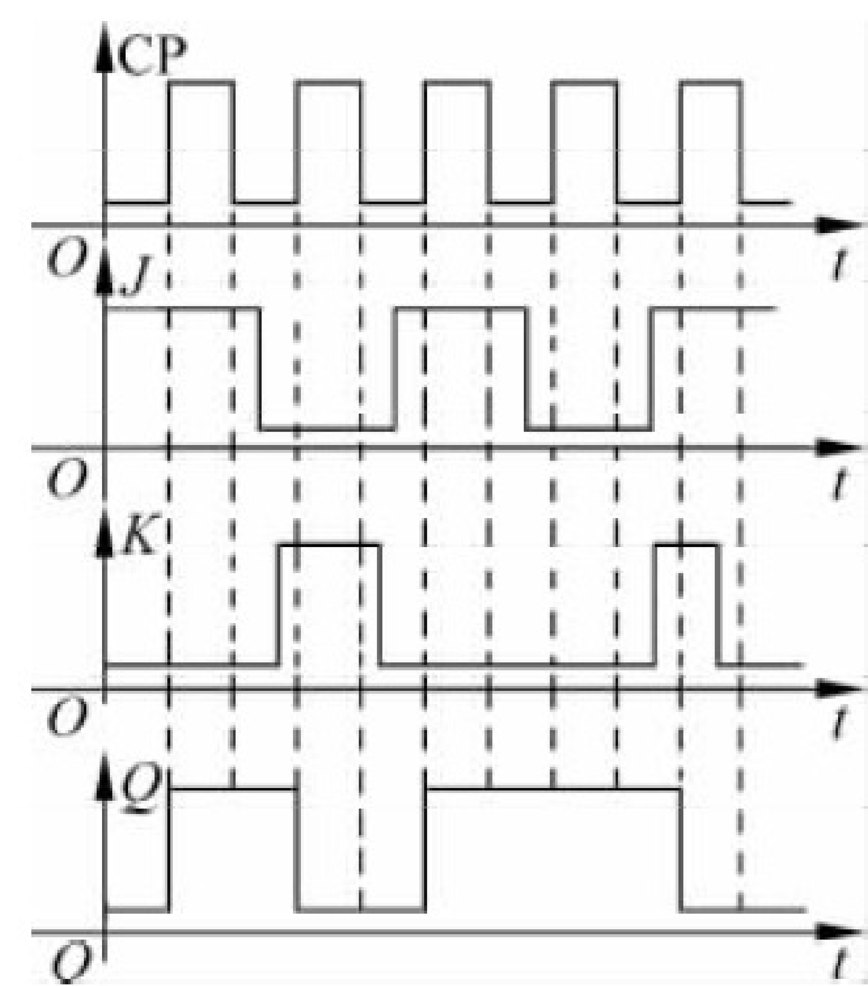
1. 特征方程:$Q^{n+1}=S+\overline{R}Q^n$ 约束条件:$RS=0$
2. 特征方程:$Q^{n+1}=D$
3. $D=\overline{\overline{J\overline{Q^n}}\cdot\overline{\overline{K}Q^n}}$
4. $J=K=M\oplus N$



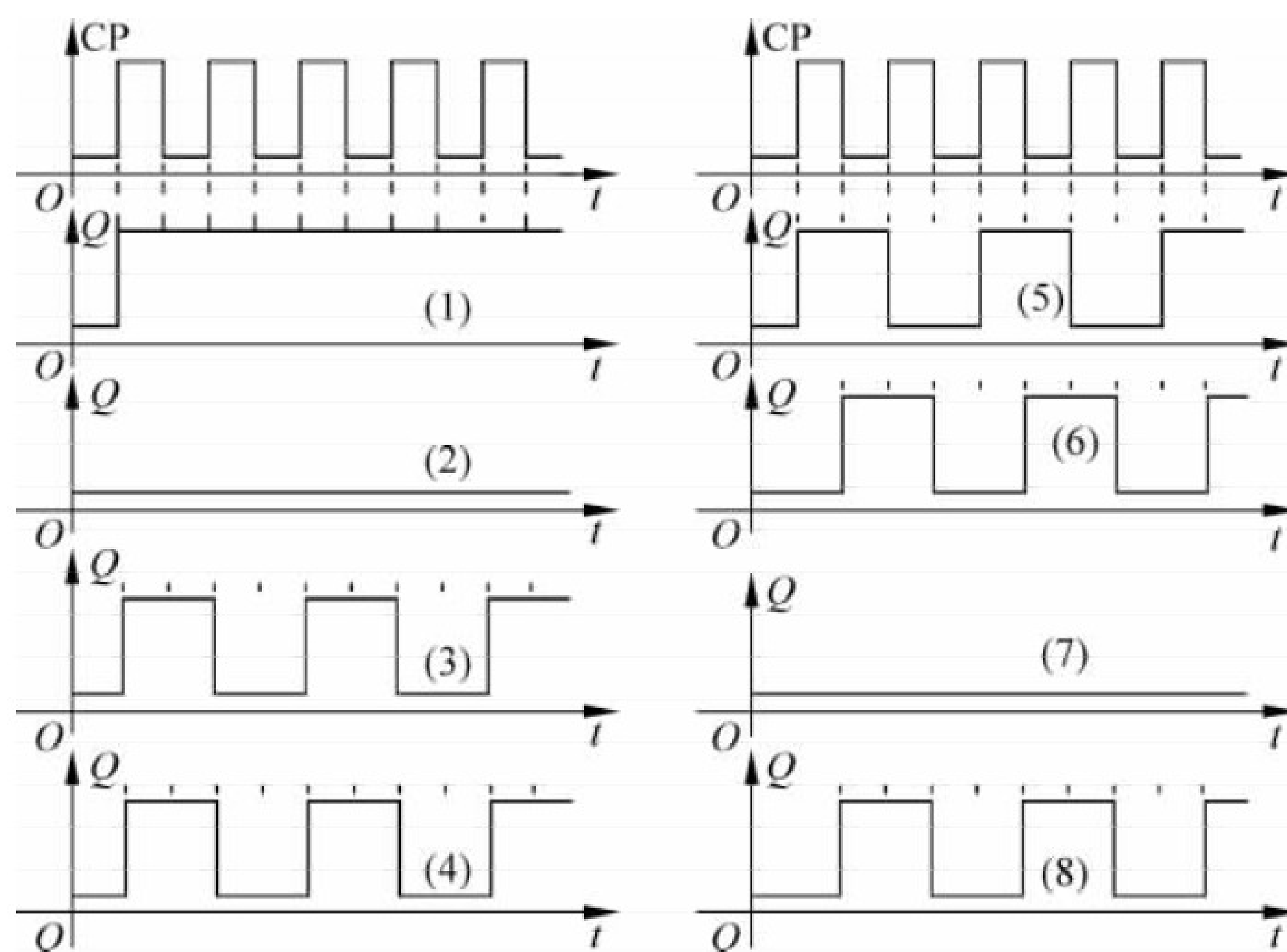
习题 5.2 5 的图



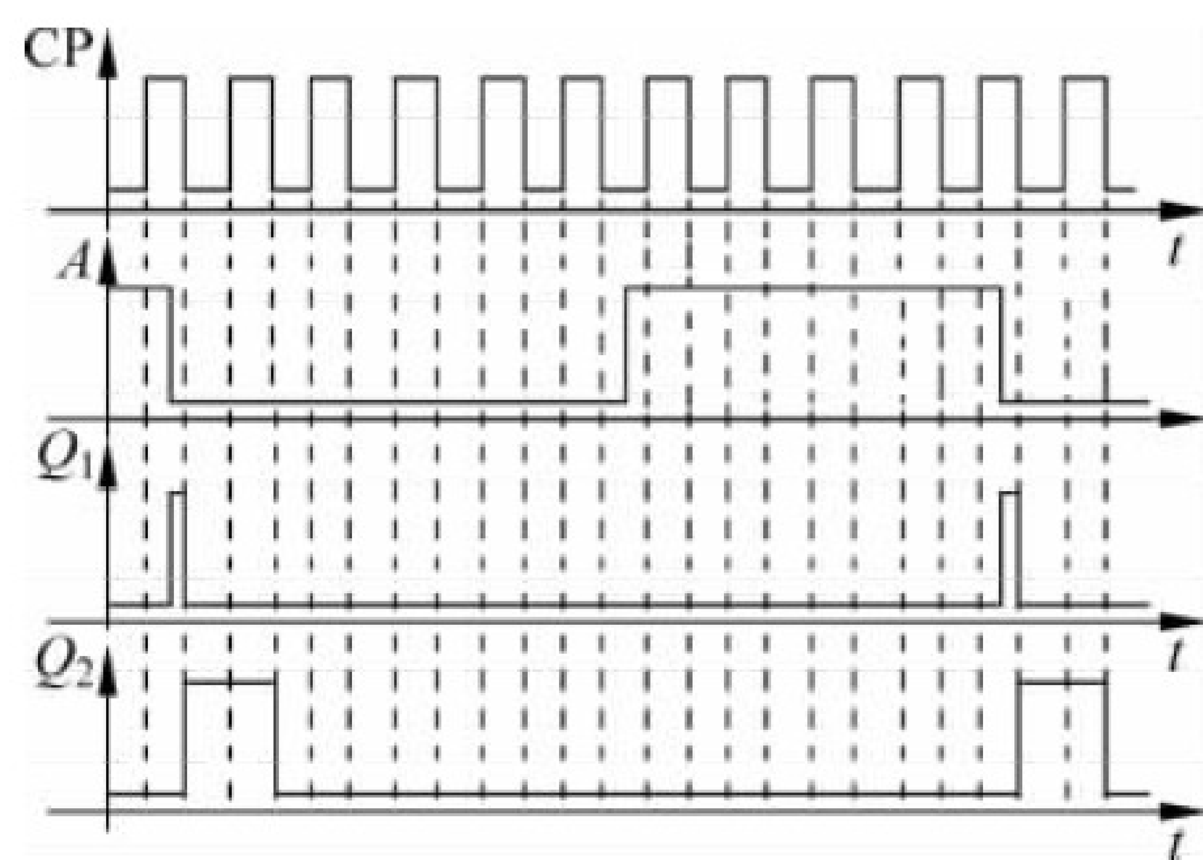
习题 5.2 10 的图



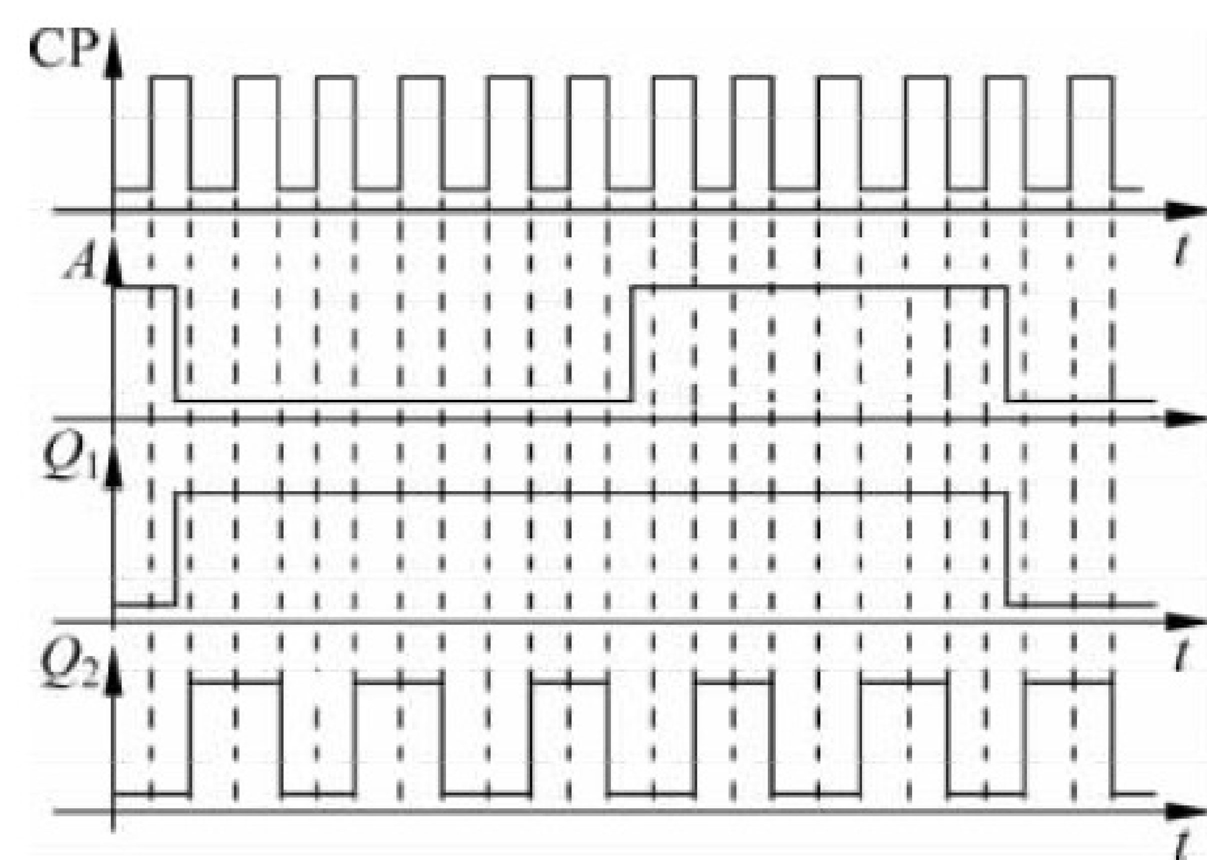
习题 5.2 12 的图



习题 5.2 13 的图



习题 5.2 16 的图



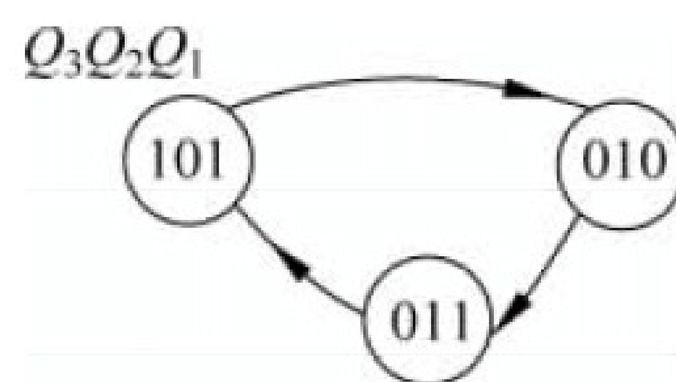
习题 5.2 17 的图

第 6 章

6-2 分析应用题

1. “111”序列检测器

2. “110”输入序列检测器



习题 6.2 3 的图

4. 三大方程如下：

   $D_1=X$， $D_2=Q_1^n$， $Q_1^{n+1}=X$， $Q_2^{n+1}=Q_1^n$，

   $Y=(Q_1^n+Q_2^n)X$

5. 状态变化规律：0 → 3 → 2 → 5 → 0

6. 状态变化规律：3 → 5 → 2 → 3

7. 可控三进制计数器，$X=1$ 允许计数

8. 4 位右移移位寄存器。

9. 异步 4 位二进制减法

10. 异步八进制减法计数器

11. 异步四进制减法计数器

12. 8 位右移移位寄存器

13. 扭环型计数器(不能自启动)

14. 能自启动的环型计数器

15. 九进制计数器

16. 十进制计数器

17. 十进制计数器

18. 九进制计数器

19. 三十二进制计数器

20. 左边的 191 构成四进制计数器，右边的 191 构成二进制计数器，为带借位输出的八进制减法计数器

21. 带进位输出的九十一进制加法计数器

22. 当 $M=0$ 时，该电路为十进制上升沿计数器；当 $M=1$ 时，该电路为十二进制上升沿计数器。

23. 二十进制计数器

24. 4 位顺序负脉冲发生器

25. 与时钟同步 4 位顺序负脉冲发生器(忽略反相器的传输时间)

26. “0101”序列发生器

27. 九进制计数器

28. 状态 S_2、S_3 重复

29. 六进制计数器

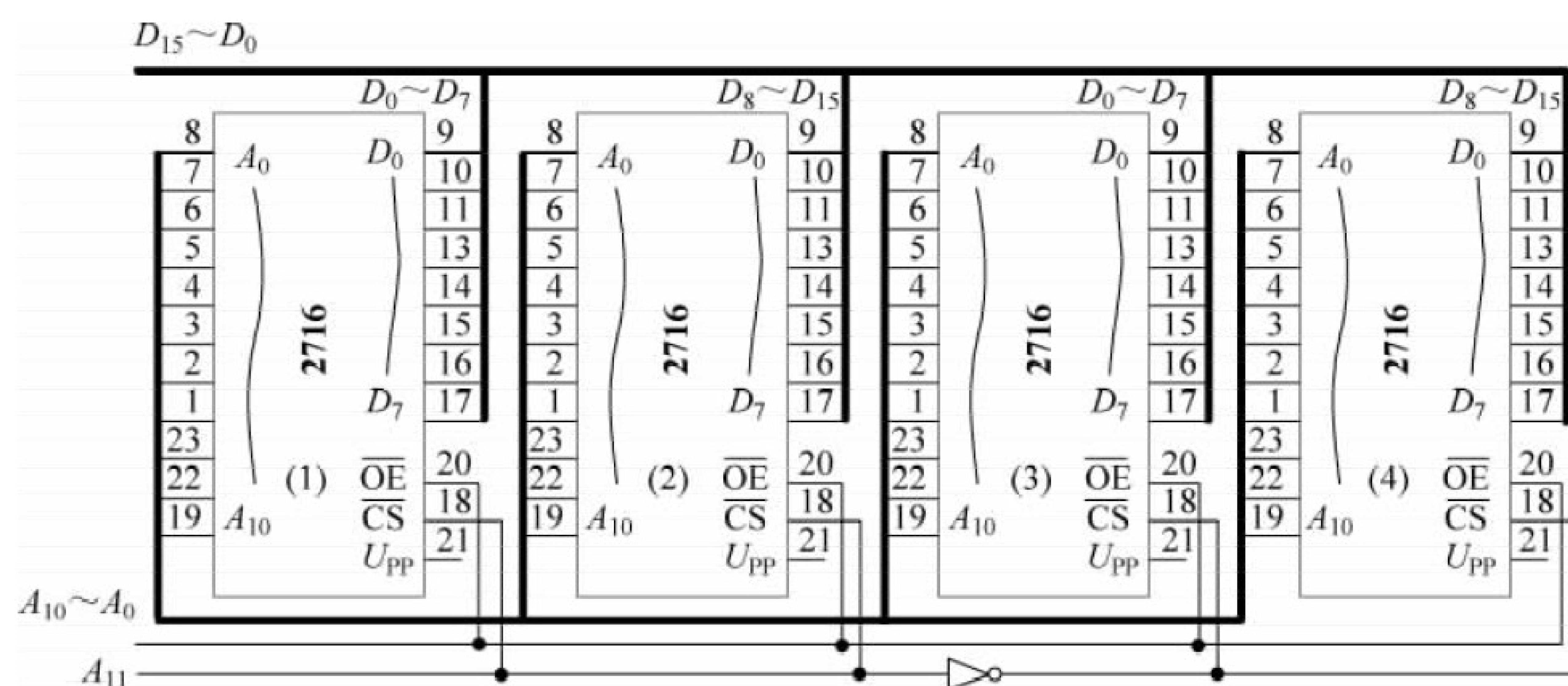
第 7 章

7-2 分析应用题

1. 4-16 的行译码器及 6-64 的列译码器

2. 8 根地址输入线、8 根字线、1 根位线；16

6. 具体如下图



习题 7.2 6 的图

7. $Y_2=AB+AC+C\overline{D}+BD+\overline{B}\overline{D}$ 或 $Y_2=AB+AC+BC+BD+\overline{B}\overline{D}$

   $Y_1=CD+\overline{A}C+\overline{B}D+A\overline{C}\overline{D}$

8. 片(1)、片(2)：0～1k－1；片(3)、片(4)：3k～4k－1

9. $D_3=\overline{A}_0+A_1$， $D_2=\overline{A}_1$， $D_1=\overline{A}_0$， $D_0=\overline{A}_1+A_0$

第 8 章

8-2 分析应用题

1. 0～7.5V

2. 12

3. 4.89mV，$\frac{1}{1023}$

4. 9.785mV，$\frac{1}{511}$，－5V

5. 7.5V

6. 5V

7. －640mV

9. 1010

10. 1010

11. 11001101

13．10101011

第 9 章

9-2 分析应用题

2．5V

3．8V、4V

4．$R_1/R_2=0.5$、$U_{DD}=10V$

5．2.277ms，10V

6．2.277ms

7．0.138ms

10．0.197Hz

11．0.152Hz

13．3nF

14．5MHz

15．0.01μs

16．6.67V，3.33V

17．6V，3V

18．0.5225s

21．144.3kHz

第 10 章

10-2 分析应用题

1．$Y_1=A\overline{B}+B\overline{C}D$， $Y_2=A\overline{C}D+\overline{A}\overline{B}\overline{D}+\overline{A}CD$

4．异或门

参 考 文 献

[1] 陈新龙.电工电子技术基础教程[M].北京：清华大学出版社，2006.
[2] 陈新龙.电工电子技术[M].北京：清华大学出版社，2008.
[3] 陈新龙.电工电子技术基础教程[M].2版.北京：清华大学出版社，2013.
[4] 陈新龙.电工电子技术基础教程全程辅导[M].北京：清华大学出版社，2009.
[5] 陈新龙.电工电子技术(上、下)[M].北京：电子工业出版社，2004.
[6] 陈新龙.电工电子技术基础教程全程辅导[M].2版.北京：清华大学出版社，2013.
[7] 阎石.数字电子技术[M].4版.北京：高等教育出版社，1998.
[8] 阎石.数字电子技术[M].5版.北京：高等教育出版社，2006.
[9] 阎石.数字电子技术[M].6版.北京：高等教育出版社，2016.
[10] 王毓银.数字电路逻辑设计[M].3版.北京：高等教育出版社，1999.